Preface

Special Issue on Mathability

The concept of mathability was introduced in 2013 as a branch of cognitive infocommunications that investigates combinations of artificial and natural cognitive capabilities relevant to mathematics, including a wide spectrum of areas ranging from low-level arithmetic operations to high-level symbolic reasoning. In its definition, it was pointed out that investigations on mathability extend to the question of how artificial mathematical capabilities can be quantified. An important goal of mathability is to develop a set of methodologies, which can be applied to emulate and enhance human mathematical capabilities.

During the last years, questions connected to this concept have been investigated by several scientists. The topics considered and discussed include, among others, connections between mathability and computer assisted methods in mathematics, the relation of some human factors to mathability, possible quantifications associated with mathability, relationships between the mathability of devices and human mathematical abilities as well as cognitive and educational aspects of mathability (e.g. mathability and cognition, computer assisted education, spreadsheet tools, coaching in education, creative problem solving). More than 60 scientific papers published in prestigious scientific journals or conference volumes as well as several chapters or parts of books have been devoted to the presentation of these studies.

This special issue extends and enriches these results with a collection of some recent achievements in this field.

Attila Gilányi

Guest Editor

Mathability in Business Education

Viktor László Takács[1] (orcid 0000-0001-8433-6115), Katalin Bubnó[2] (orcid 0000-0002-5445-2374)

[1] Department of Business Informatics, University of Debrecen, Böszörményi u. 138, 4032 Debrecen, Hungary; takacs.viktor@econ.unideb.hu

[2] Doctoral School of Mathematical and Computational Sciences, University and National Library, University of Debrecen, Egyetem tér 1, 4032 Debrecen, Hungary; kbubno@lib.unideb.hu

Abstract: Data Analyst or Data Scientist is one of the most sought-after professions. In tertiary education, the most common way for someone to become a Data Analyst is to learn Business and, later, if someone has some computational skills they can learn Computer Science as well. With the two degrees, they can solve real-life problems in their job. But there are a lot of necessary things they had never studied before because the basic conceptual methodology and the mindset were not based on any of the two curricula. Our aim is to develop a method in introductory business education to lay down the bases of the right-thinking towards data sciences for undergraduate students in Economics. If they are interested in the topic, they can orient themselves towards data sciences during their further studies. In this paper, we present a methodology to help Economics and Business Students to understand the main business questions, formalize the questions correctly, find computer-based solutions, and discuss/debug the results. Furthermore, students must learn basic data transformations and data enrichment methods as well, which is the primary feature of high-mathability problem-solving approaches. The tool we use is Microsoft Excel which contains OLAP (On-Line Analytical Processes) elements.

Keywords: OLAP; data science; formalization; management questions; computer problem-solving; high-mathability teaching approaches; knowledge transfer; spreadsheets

1 Introduction: Computational Thinking and Data Science

We aimed to develop three critical skills for undergraduate students in economics so that, if they are interested in the topic, they could orient themselves towards data science during their further studies. The three skills are problem-solving, analytical thinking, and system approach. These are all part of computer thinking [9]. The term Computational Thinking was first used by Seymour Paper [5], but it has become an important research area for IT methodology over the past ten years [4]. Many definitions have been created, reinterpreted, and explained, but everyone agrees that computational thinking should not only be a matter for IT professionals, but a key competence for everyone.

Our first and most important skill is problem-solving, without this, the data scientist cannot be good in his profession. This skill is not technology-dependent, problem solving is developed mainly in mathematics classes in public education. Typical tools are mathematical word problems.

In a previous study [7], we discovered some analogies between mathematical problem-solving and the process of computer programming based on Pólya's problem-solving method [6]. We worked out a teaching methodology introducing computer programming. We had early developing courses in a primary school, and later we created an experiment in a secondary school [7, 10], and we presented some cognitive aspects of our experiments and results [26] based on Ambrus's mathematical psychological research [27]. Meanwhile, we start to teach Business Informatics for Economics and Business Students. We recognized the similarities to word problems and we adapted Pólya's method to solve business questions as domain-specific word problems with OLAP (On-Line Analytical Processing) technology.

Based on Pólya's problem-solving method [6] a Data Analyst must go through some steps:

- Understanding the problem,
- Devising a plan,
- Carrying out the plan,
- Discussion.

In this paper, our goal is to show an introductory method and tools to help Economics and Business Students to understand usual business questions, formalize the questions correctly, solve and discuss/debug the result. The truth is that the more serious discussion will be the topic of further courses (i.e. Information systems or Risk analysis).

The second skill to be developed is analytical thinking, while the third is the system approach. Developing the right system approach is difficult because students think that they have to work with prepared data tables and they only have to master the technical processing. We believe that students coming to universities should be taught a completely new approach to informatics, instead of teaching the office user knowledge that they should have acquired in high school, we can base their confident technical knowledge and system approach. Unfortunately, high school data processing skills can even cause misconceptions for students. There are no large amount of data management tasks in high school informatics studies in Hungary, neither in spreadsheet nor database topics. Students need to be faced with large datasets as soon as possible to process data that is impossible to do manually one by one at the university level. To do this, they need to learn effective methods and approaches.

2 Mathability in Business Education

The first time we start the work with students we make it clear that there are two types of information systems, the first is transaction-oriented (OLTP) systems, and the other is analytics-oriented (OLAP) systems. The difference is the aim and the structure of the system. When we want many workers to work in a system simultaneously and input data about the operation of an enterprise, we use transactional (OLTP) systems. When we want to analyse the formerly recorded data we use analytical (OLAP) systems. The structural difference is the topic of a further course (Development and Management of System of Information systems), in this course we focus on the question of when and why we use the two types.

Usual problems for beginner Data Analysts are to create the visualization, and report based on a dataset (aka. unfolded OLAP cube). Microsoft Excel also contains OLAP elements, a fact that has greatly democratized data science, allowing anyone to learn the basics of OLAP through one of the most well-known application software. To achieve the goal they must learn basic data transformation and data enrichment methods, too.

High-level knowledge of spreadsheets is necessary in the economic field and this is more and more the OLAP knowledge [3] when analyzing job advertisements for data scientists. We found and adapted an appropriate technique (Sprego) to achieve our goals in Maria Chernoch and colleagues' Sprego work [1, 11, 12, 13, 14]. The essence of the Sprego method is that even though Excel has more than 600 built-in functions, we use up to a dozen of them regularly [8]. Starting from this basic idea, the Sprego method has defined the basic function group that we can use to solve many tasks in spreadsheets, using them as building blocks.

In the sense of CogInfoCom or mathability the other parts of our topics were investigated in some studies. We mentioned Csernoch and colleagues' spreadsheets research.

In 2012 Baranyi and Csapó provided the finalized definition of CogInfoCom and provided an overview of a number of related fields [15]. In the 2013 CogInfoCom Conference, where the concept of Mathability has been defined Gilányi and Baranyi declared that important goals of mathability are to investigate how artificial mathematical capabilities can be quantified, and to develop a set of methodologies using which human mathematical capabilities can be emulated and enhanced [16]. Gilányi and colleagues presented the practical usage of the mathability concept approach via tertiary education content many times [21, 22, 23, 24, 25]. Furthermore, there were several experiments in secondary education in problem-solving domain emphasizing the strong connection between mathematical and computational thinking [28, 29].

In Data Science – as we call it nowadays business analyses supported by computer programming and artificial intelligence – is a very popular domain in business and industry. Tertiary education can hardly follow the requirements of the market, and there are no suitable teaching methodologies to lay down the basics of the right conceptions of the topic. It is alarming that there are neither teaching nor evaluating methods for selecting the persons who could become really good data scientists.

Further important findings were found regarding constructive learning methods in Chmielewska and Gilányi's papers [17, 18, 19], whose experiments focused on computer-assisted self-study, furthermore, Chmielewska and Matuszak developed a coaching method as well [20].

Obviously, in many areas, it is essential today to lay the foundations for data processing and OLAP technology for example in medical, engineering sciences [30, 31].

3 Teaching Methodology in Introductory Business Informatics Courses

We divided the lessons into five parts:

1) Introduction to Excel usage,
2) Introduction to formulas,
3) Introduction to analysis,
4) Analytics,
5) Analysis with Pivot Tables and dynamic charts.

3.1 Introduction to Excel Usage[1]

Our first problems are the technical studies of excel usage. At first sight, it does not seem to be a mathematical capability. However, Csernoch [1, 11] investigated this problem as well because it can refer to the conscious usage of the spreadsheet tool. She declared the trial-error or bricolage level when a user was obviously just clicking back and forth without a conscious reason.

Chmielewska and colleagues pointed out [17, 18, 19], that uncontrolled computer-assisted self-education can be risky, while a lack of accuracy and sketchy solutions are characteristic for the young generation. Furthermore, they showed how important it is to reflect on the results obtained while learning by trial and error because the lack of feedback can lead to misunderstandings and misconceptions.

[1] The exercise location: http://takacs-viktor.hu/pages/excel/Xcelpract2016_TVL.xlsx Worksheet: 'Basics'

We agreed with this, and we begin the course by showing students what their former knowledge (including technical management of Excel) is about when we started to develop the computational tool consciously.

We believe the usage could be taught as seriously as a ‘real’ mathematical problem because it depends on analytical thinking. Instead of clicking back and forth, and instead of guessing what the program might know, the user has to click consciously and discover the functionalities of the application software. For that, the user has to recognize the same scheme as when they start to discover a computer tool.

In this domain, common problems are manipulations with various excel objects. There are no strict differences between the plans in these example areas. So this ’tool’ is good for introducing the method.

We have a prepared excel practice worksheet in which we teach the basic introductory manipulation examples with.

The tasks of the first exercise:

1) Unhide the hidden columns and rows,
2) Delete the columns with headers ’ab1’, ’acc’, ’bb1’, ’bcc’,
3) Search the cell which has content ’xxxxxxxxxx’,
4) Format the columns A, D, E to the same width,
5) Copy this worksheet and rename it to ’Data’ and delete the columns A-G,
6) Insert a new (empty) first row,
7) Insert a new column B and hide it.

Even in the introduction to excel and formalization, we follow Pólya’s steps.

In our former research study, we detected some analogies between Pólya’s steps and the steps of algorithmization. We used it in novice programming environments to teach the basics of computer programming, but as Csernoch and colleagues pointed out, the language of spreadsheets are functional programming languages [1], so we can transfer our methods into spreadsheets as well in this introductory course that we can see in Figure 1. [7]

Word problems in mathematics	Computer programming
1. Comprehension, understanding the problem	
2. Devising a plan	
Determinantion of unknown variable	Declaration of variables
Mathematical model (plan)	Algorithms and their implementation (plan)
3. Carrying the plan	
Solving equations	Compiling and running
Answering to the questions of the starting problem	
4. Discussion (Looking back)	
changing initial conditions	testing
generalizing the problem	debugging
looking for other methods for solution	optimizing code

Figure 1

Analogies between Pólya's steps, algorithmization, and common spreadsheet technical manipulations

3.1.1 Pólya's First Three Steps

The first step is understanding the problem, followed by detecting necessary data and declaring variables.

In the problems (tasks) described above, we can recognize four types of variables: function, object (Excel), condition, and parameter. It can be useful if we form all the tasks in a table, like in Table 2:

Table 1

The variable recognition in the first exercise

function	object (excel)	condition	parameter
unhide	column	visible = false (hidden)	width = …
hide	row	content = 'xxxxxxxxxx'	number format = …
delete	cell	id = A	height = …
copy	worksheet	id = 1	
insert	workbook	name = 'Data'	
select	selection	header = 'ab1'	
search	null		
format			

We have a special object named as selection, which is a conditional object, a subset of an excel object, and three special functions:

- select usually based on a condition, the result is a selection,
- search is a specialized 'select' based on a condition also, the result is a special selection, in this case a range of cells,

- the format usually has parameters (set an attribute of an object to a value) and the result is empty.

In this domain condition is a selection criterion, it contains an attribute of an object, a value, and a relation (usually the equal relation).

Formerly, in solving mathematical word problems we created a mathematical model and made the algorithm and their implementation in the concrete programming environment when we taught it as a novice computer programming task. Here, when we create the generalized steps (carriable logical plan) to solve the domain-specific ’word problems’, similar to database design methodology [2], the following three steps are essential in the design of data warehouses and executive information systems, which are enhancements of Pólya’s ’Devising a plan’ step:

- Conceptual planning,
- Logical planning,
- Physical planning.

Conceptual planning means that ‘I know what to do with which object and by what condition or parameter’:

1) select an excel object,
2) apply a function on the selection,
3) set the possible parameters.

Logical planning means the formalizing of the concrete tasks one by one and translating the logical plans to physical plans. Carrying out the plan in these cases gives immediate results as soon as we do the manipulation on the object.

In the first task, we can see that we must divide the problem into two plans, because we realize that we cannot select hidden columns and rows, so we must select column and row ranges where the selection borders are visible columns and rows.

1) unhide(select(column, visible=false))
2) unhide(select(row, visible=false))

When we translate the logical plan to a physical plan we simplify the problems.

1) unhide(select(column, id ϵ {B,..,F})),
2) unhide(select(row, id ϵ {5,..,20})).

Logical plans of the second task:

1) delete(select(column, search(cell, content=’ab1’))),
2) delete(select(column, search(cell, content=’acc’))),
3) delete(select(column, search(cell, content=’bb1’))),
4) delete(select(column, search(cell, content=’bcc’))).

In this case we have A, B, C, D conditions which define four selections that could be described with one complex condition E = A or B or C or D = A + B + C + D, the result will be the logical plan:

delete(select(column, search(cell, content=’ab1’ or content=’acc’ or content=’bb1’ or content=’bcc’))).

The physical plans for each logical plan are as follows:

1) delete(select(column, id=G)),
2) delete(select(column, id=H)),
3) delete(select(column, id=I)),
4) delete(select(column, id=j)).

or delete(select(column, id ϵ {G,..,J})).

From tasks 3 to 7 the plans are:

1) search(worksheet, content = ’xxxxxxxxxx’),
2) format(select(column, id ϵ {A,D,E}, width=a), where ‘a’ is a parameter value,
3) this is a complex problem again:
    a. copy(worksheet, this),
    b. rename(worksheet, new, name=’Data’),
    c. delete(select(columns, id ϵ {A,..,G})),
4) insert(row, 1, 1).
5) another complex problem:
    a. insert(select(column, id=B), 1),
    b. hide(select(column, id=B)).

3.1.2 Looking Back

When we discuss all the above, we can create the generalized formal structure: (f,o[,c][,p])->o, basically f(o[,c][,p])->o

object=function(object[,condition][,parameter]).

We can develop an excel object hierarchy also: cell -> column and row -> worksheet -> workbook.

The Excel object hierarchy is important when we specify an Excel object, every definition contains every mother level of an excel object, when we leave any hierarchy level ’unspecified’, that means ’actual’ or ’this’ worksheet or workbook of the cell, row or column. This object hierarchy is very important when cell references occur in later physical plans.

We extend the introductory manipulation in several ways, and always refer to the discussed introductory formal structures e.g. create function, table, pivot table, chart, slicer objects, with the relevant task, problems also, e.g. create a pivot table from the dataset, create a column chart based on a pivot table, set pivot table connection to a slicer, etc.

3.2 Introduction to Formulas[2]

From this point, we use the fully extended Pólya's steps of planning.

The identified variables of formulas as a problem-solving domain are result and formula in the following structure: result = formula, where the equal symbol must be written directly into the result selection, and formula must be replaced with the logically planned solution.

Basic introductory formula exercises:

1) Calculate Total Cost and format as Forint (Ft)
   a. hourly cost: 1,000 Ft/h
   b. Nr of hours: {23 h, 34 h, 45 h, 56 h, 78 h}
2) Fill the multiplication table!
   a. a: {1, 2, 3, …, 10}
   b. b: {1, 2, 3, …, 10}

3.2.1 Understanding the Problem

As you can see in the above examples we have constants, arrays, and matrices.

3.2.2 Devising a Plan

Conceptual plans: result = formula.

Logical plans:

1) There are two possible logical plans:
   a. A natural logical plan: $\overline{TC} = C * \overline{H}$, where TC is a Total Cost result array, C is the Cost constant, and H is the number of hours array.
   b. An alternative logical plan: $TC_j = C * H_j, 1 \le j \le 5$, where j is the j^th^ element of a related array.
2) There are two possible logical plans:

2 The exercise location: http://takacs-viktor.hu/pages/excel/Xcelpract2016_TVL.xlsx Worksheet: 'Formulas'

a. A natural logical plan: $\overline{M} = \overline{a} \times \overline{b}$, where M is the result multiplication matrix, a and b are arrays with n elements.

b. An alternative logical plan: $M_{ij} = a_i * b_j, 1 \le i, j \le 10$, where i and j are the i^{th} and j^{th} elements of a related array and/or related matrix.

Physical plans:

In the process of transforming the logical plans to physical plans we define the excel reference objects (cell or cells) in both the result and formula parts. Generally, in our examples, the alternative logical plans are easier to implement, but the natural logical plans are more sophisticated solutions, and the plan will be closer to the natural thinking method.

In our examples the i translated to row identifier (numbers), j translated to column identifier (letters).

1) In this example we translate the logical plans to physical plans in two ways:

   a. Cell range based: $\overline{\boldsymbol{TC}} = \boldsymbol{C} \times \overline{\boldsymbol{H}}$ →B7:F7{=B5*B6:F6} the result planned into a cell range, the formula must be accepted with [CTRL]+[SHIFT]+[ENTER] key combination.

   b. Cell-based: $TC_j = C * H_j$ →B8=$B$5*B$6, (j=1)

      $1 \le j \le 5$ →copy(cell,B8,B8-F8)

      the result panned into a cell, we should fix the row part of the cell references, and the column part must remain in relative format, and we must copy the cell B8 (formula) from the result cell B8 until the cell F8.

2) In this example we translate the logical plans in two ways:

   a. Cell range based: $\overline{M} = \overline{a} \times \overline{b}$ →C14:L23{=B14:B23*C13:L13} the result planned into a cell range, the formula must be accepted with CTRL+SHIFT+ENTER key combination.

   b. Cell-based: $M_{ij} = a_i * b_j$ →C14=$B14*C$13, (i,j=1)

      $1 \le i, j \le 10$ →copy(copy(cell,C14,C14:L14),C14:L14-C23:L23)

      the result planned into a cell, we must use partly relative formula according to the **a** and **b** arrays in the logical plan, and we must copy the cell (formula) both between the columns and the rows.

3.2.3 Carrying Out the Plan

Generally, when we carry out the physical plan, we do the following steps:

1) select the result object (cell or cell range), green part of the plan,
2) write the formula, the black part of the plan, starting with =,
3) accept the formula with ENTER or CTRL+SHIFT+ENTER,
4) copy the result if you must.

We have to emphasize many times that as we have more complex problems, the number of logical plans could be increased. There is no single good solution. There are many possible ways of thinking and many possible logical plans. We must choose from them which one to solve. In the sense of mathability, when we examine humans and the computer as a whole system, when we want to develop computational thinking we have to highlight points when human thinking or decision is essential and cannot be substituted with artificial, emulated skills (or at least not at this level).

3.3 Introduction to Analysis[3]

In this domain, we are working with unfolded OLAP cubes (aka datasets), where OLAP means the classical On-Line Analytical Processes, and our database are organized to serve analytical needs. A dataset contains only two types of data: Measure and Attribute. Measures are always numbers with a unit, Attributes are everything else.

3.3.1 Understanding the Problem

The most basic task is to show a projected attribute or aggregated measure on a selected subset of an unfolded OLAP cube. This is the step when we recognize relevant data and gather them as variables.

Table 2
Analytical variable recognition

Recognized variable	Type of the variable		Relational algebra analogy
aggregate function	result		aggregation
Attribute xor Measure	result		projection
Condition	condition		selection

We must gather the variable values from the problems, choose a formula and substitute the values in the formula. Depending on the students' knowledge and interest we can introduce the basic concepts of relational algebra to lay down the foundations for future database management as we can see it in Table 2.

3.3.2 Devising a Plan

Conceptual plans in this domain are grouped in two basic structures:

3 The exercise location: http://takacs-viktor.hu/pages/excel/Xcelpract2016_TVL.xlsx Worksheet: 'AnalysisIntro'

1) Conditional aggregation (1), means an aggregation of a measure where C condition is TRUE.

$$result\{= af\left(IF\left(C, M\right)\right)\} \quad (1)$$

2) Conditional select (2), the result is an attribute or measure value, where the condition is first time TRUE

$$result = INDEX\left(A/_{M}, MATCH\left(C\right)\right) \quad (2)$$

For these two basic structures, we must define the elements:

- ***result*** is the projection of an attribute or an aggregated measure,
- ***aggregate function*** is an element of the set of basic functions {SUM(), MIN(), MAX(), AVERAGE(), and COUNT()}. This set can be extended with special LARGE() and SMALL() and statistical aggregations too. note: COUNT() usually planned and implemented with a SUM() function of a measure 1,
- ***condition*** is a three-tuple consisting set (3), an attribute or measure is in a relation with a value.

$$c \in C, A/_{M} \begin{matrix} < & \geq \\ > & \leq \\ = & \neq \end{matrix} V \quad (3)$$

On a logical level, we refer to the cubes' measures and attributes as a systematic arrangement of values, noted by the name of the attribute or the measure.

When we translate logical plans to physical plans, we must be aware of some considerations related to datasets. Every column of a dataset is a measure or an attribute. The first row must contain the name of the measures and the attributes, and values from the second row to the last. It must not be an empty column and/or row in the dataset. Dataset is a set of column-based cell ranges.

In the process of transforming the logical plans to physical plans, we define the excel reference objects (cell or cells) in both the result and formula parts. Usually, it must not but should be a good practice to fix the data source references of measures and attributes.

Additionally, we can define named tables with attribute and measure columns over a cell range as a column-based dataset. With these additions, our physical plans get closer to the logical plan.

To achieve this, we must follow the plan:

1) select(cell, A3:I102),
2) format(selection, table, header=true),
3) rename(table, name='dataset'),

4) select(cell, L18:N23),
5) format(selection, table, header=true),
6) rename(table, name='BMI').

When we plan complex problems, we follow the top-down planning method to break down a complex problem into simple sub-problems.

3.3.3 Carrying Out the Plan

Generally, when we carry out the physical plan, we build up our solution formulas strictly following the bottom-up concept, starting with the formulas related to the independent simple sub-problems.

3.3.4 Basic Introductory Analysis Examples

What is the pulse of the 51-year-old patient? - we have no defined aggregation in the task. In this case we, can use either the conditional aggregation or the selection conceptual formulas.

1) Conditional aggregation, we can use any aggregation from the four basic ones, because we get back a one-element-array of Pulse as a result of the condition.

Table 3
Variable recognition of conditional aggregation

Understanding the problem	af	any from {MIN, MAX, SUM, AVERAGE}
	A/M	Pulse
	C	Age = 51
Conceptual plan		$result\{= af(IF(C,M))\}$
Logical plan		$result\{= SUM(IF(\overline{Age} = 51, \overline{Pulse}))\}$
Physical plan		K5={SUM(IF(B4:B105=51,I4:I102))}
Physical plan based on tables		K5{=SUM(IF(dataset[Age]=51,datase[Pulse]))}

2) Conditional select, where 51 is the value from the condition, Age is the attribute part, and the last 0 is the equal relation.

Table 4
Variable recognition of conditional search

Understanding the problem	af	-
	A/M	Pulse
	C	Age = 51
Conceptual plan		$result = INDEX(A/M, MATCH(C))$
Logical plan		$result = INDEX(\overline{Pulse}, MATCH(51, \overline{Age}, 0))$
Physical plan		K5=INDEX(I4:I102,MATCH(51,B4:B102,0))
Physical plan based on tables		K5=INDEX(dataset[Pulse],MATCH(51,dataset[Age],0))

3.4 Analytics[4]

We are working with OLAP technology, where the data is stored in OLAP cubes with one or more fact tables and dimensions. Fact tables contain aggregated or raw measures and special attributes as dimension keys, dimensions have unique identifiers as dimension keys and dimensional attributes and possible dimension hierarchies. The fact tables and dimensions are connected to each other through the dimension keys.

The usual conceptual problems in this domain are:

- Data enrichment and calculated measures,
- zero to two-dimensional details of an aggregated measure,
- unfold dimensional attribute and/or measure into the cube.

Analysis examples

1) Data enrichment:
   a. Extend the date with temporal dimension attributes.
   b. Calculate the Quarters or Percentiles of the observation.
2) Simple answers:
   a. Calculate the sum of the ordered units.
   b. Which is the largest number of units in the orders?
   c. How many representatives got gifts?
   d. How many gifts did Quebec region get?
3) One dimensional aggregation:
   a. How many orders are there by regions?
4) Two dimensional aggregation:
   a. How many pieces did the representatives buy per Item?
   b. Compare the Cost (HUF) detailed by regions, by two freely selectable values of Item.
5) Unfold dimensional attribute:
   a. Calculate the Unit Price ($) for every order based on the item's monthly price list.
   b. Calculate the Cost (HUF) based on daily exchange rates.
   c. Specify currency for every region in the orders.
6) Calculated Measure:

[4] The exercise location: http://takacs-viktor.hu/pages/excel/Xcelpract2016_TVL.xlsx Worksheets: 'Orders', 'Prices' and 'Rates'

a. Calculate Cost ($) in every order.

b. Calculate the Delivery time.

3.4.1 Devising a Plan

Conceptual plans in this domain are grouped into two basic structures.

1) Conditional aggregation with zero to n details (4), this means an aggregation of a measure where C condition is TRUE.

$$M([A_1].[,A_n])\{=af(IF(C,M))\} \quad (4)$$

2) Unfold dimensional attribute or measure into the fact table (5).

$$factA/factM = INDEX\left(\dim.A/\dim.M, MATCH\left(fact.DK_i, \dim.DK, 0\right)\right) \quad (5)$$

$$1 \le i \le n, n = |fact.DK|$$

The result is an unfolded attribute or measure value, where the fact table dimension key is found in the dimension unique key, based on the many-select problem.

Condition is a three-tuple consisting set, an attribute or measure is in a relation with a value.

We use boolean simple logic structures (6), translated to physical plan.

$$a,b,c,d \in C, c = a \wedge b = (a)\times(b), d = a \vee b = (a)+(b) \quad (6)$$

3.4.2 Examples

1) Simple, introductionary problems

Table 5

Variable recognition of simple problems

Problem		Calculate the sum of the ordered units!	Which is the largest number of units in the orders?
Understanding the problem	af	SUM	MAX
	A/M	Unit	Unit
	C	-	-
Conceptual plan		$result\{=af(IF(C,M))\}$	
Logical plan		result=SUM(Unit)	result=MAX(Unit)
Physical plan		X1=SUM($N$2:$N$53)	X6=MAX($N$2:$N$53)

2) How many representatives got gifts?

Table 6

Variable recognition of a conditional simple problem

Understanding the problem	af	SUM	
	A/M	1	
	C	Gifts<>""	Gifts>0
Conceptual plan		$result\{=af(IF(C,M))\}$	
Logical plans		result{=SUM(IF(Gifts<>"",1))} result{=SUM(IF(Gifts>0,1))}	
Physical plans		X11{=SUM(IF($Q$2:$Q$53<>"",1))} X11{=SUM(IF($Q$2:$Q$53>1,1))}	

This question is a bit confusing. How many is a simple sum of 1, but what does getting a gift mean? The natural translation should be gift = 'anything', where we have a value 'anything', but this is confusing as a value, we can't write 'anything' in a cell. One possible solution could be to double negate the logical expression, which means not equal as opposite of equal, and 'nothing' as the opposite of 'anything'.

3) How many gifts did Quebec region get?

Table 7

Variable recognition of a conditional simple problem

Understanding the problem	af	SUM
	A/M	Gifts
	C	Region="Quebec"
Conceptual plan		$result\{=af(IF(C,M))\}$
Logical plans		result{=SUM(IF(Region="Quebec",Gifts))}
Physical plans		X18{=SUM(IF($J$2:$J$53="Quebec",$Q$2:$Q$53))}

4) How many orders are there by regions?

Table 8

Variable recognition of a conditional simple problem

Understanding the problem	af	SUM
	A/M	1
	C	Region=actRegion
Conceptual plan		$result\{=af(IF(C,M))\}$
Logical plans		result{=SUM(IF(Region=actRegion,1))} ,where actRegion = $Region_i \in Region$, $1 \leq i \leq n$, n=|Region|
Physical plans		X18{=SUM(IF($J$2:$J$53=$V18,1))} copy(cell,X18,X18-X20)

5) How many pieces did the representatives buy per Item?

Table 9
Variable recognition of a conditional simple problem

Understanding the problem	af	SUM
	A/M	Units
	C	Representative=actRepresentative AND Item=actItem
Conceptual plan		$result\{= af(IF(C,M))\}$
Logical plans		result{=SUM(IF((Representative=actRepresentative)*(Item=actItem), Units))}, where actRepresentative = $Representative_i \in$ Representative, $1 \leq i \leq n$, n=|Representative|, actItem = $Item_j \in$ Item, $1 \leq j \leq m$, m=|Item|
Physical plans		AB24{=SUM(IF(($L$2:$L$53=$AA24)*($M$2:$M$53=AB$23), $N$2:$N$53))} copy(copy(cell,AB24,AB24-AF24),AB24:AF24-AB54:AF54)

3.5 Analysis with Pivot Tables and Dynamic Charts[5]

We are working with OLAP technology, where the data is stored in unfolded OLAP cubes with one or more fact columns and dimensional columns. Fact columns contain aggregated or raw measures, dimensional columns have only dimensional attributes and possible dimension hierarchies.

We collect and formalize management questions. The question and description are defined in a textual and formal way. The 'manager' is the person for whom the system provides information. As a result, she expects to see a data visualization report appearing on different dashboards.

We 'analyse' the management question based on the following considerations in Table 10:

What is the indicator? In which aggregation? What is the unit? Which visualization do we want to see? In Which detail(s) is there a slicer?

Table 10
Management question analysis

Indicator	$\boldsymbol{I}_{\{af[,af]\}}^{\{u[,u]\}}$	the indicator ***I*** to be produced with *u* unit(s) in the upper right index and af aggregate function(s) in the bottom right index,
unit(s)		
aggregate function(s)		

5 The exercise location: http://takacs-viktor.hu/pages/excel/Xcelpract2016_TVL.xlsx Worksheet: 'PivotAndCharts'

visualization	$\left(\begin{matrix} vt \\ \left[\left[\begin{Bmatrix} s \\ s \end{Bmatrix}\right]\right] \end{matrix}\right)^{v}$	the *v* visualization with the type *vt* (table, line diagram, bar graph, etc. ...) and optional *s* slicers (values can be $\boldsymbol{D}_{\{a\}}$ dimensional attribute, $\boldsymbol{D}_{\{v\}}$ subset of concrete values, or a $\boldsymbol{D}_{\{a\}}$ dimensional attribute in the *d* detail of another ***I*** indicator on the same dashboard)
slicer(s)		
detail(s)	$\left[\left(\begin{matrix} D_{\sum\{a\}} \\ \left[D_{\sum\{a\}}\right] \end{matrix}\right)^{\{d\}}\right]$	*d* details with $D_{\{a\}}$dimensional attribute(s), with optional $\sum\{a\}$ aggregation. *d* values e.g.: *row*, *col*umn, *cat*egory, *y* indicator

Formally (7):

$$\begin{matrix} I^{\{u[,u]\}}_{\{af[,af]\}} \\ \left[I^{\{u[,u]\}}_{\{af[,af]\}}\right] \end{matrix} \left(\begin{matrix} vt \\ \left[\left[\begin{Bmatrix} s \\ s \end{Bmatrix}\right]\right] \end{matrix}\right)^{v} \left[\left(\begin{matrix} D_{\sum\{a\}} \\ \left[D_{\sum\{a\}}\right] \end{matrix}\right)^{\{d\}}\right] \tag{7}$$

Physically we must create a Pivot Table visualization when we only need to create the definition, and a Chart if needed with slicers.

Table 11

Structure of the physical definition

Filter [Filter]	Column detail [Column detail]
Row detail [Row detail]	aggregate function(Measure) [aggregate function(Measure)]

Aggregate function and measure are defined before. Filters, Slicers and Details are based on Conditions.

Conclusions

Summarizing, formerly, with using Blockly Code and our analogy-based introductory computer programming teaching method we successfully taught and automatized a problem-solving strategy that pupils could store in their long-term memory and it can help them to analyse, formalize and generalize the problem successfully. In this paper, we presented that this concept could be adopted in tertiary education also in a more serious way and higher level. In tertiary business education, the aim was to make the problem of solving management questions independent from the computational tool we use. In traditional computer science teaching (at any level) we show the tool, and we show what it is for. We use examples to present the knowledge and the borders of the computational tool we teach. But now, we taught problem-solving, and we gave the students a way of thinking that could be the successful whatever computational tool we use for solving the problem. We gave them structures or schemas for formalizing management questions when they just create pivot analyses or even develop them into data warehouses. Furthermore, we showed that we can also teach technical manipulations as they would be ‘problems’ in a conscious way and we gave students transferable knowledge with it because the model that shows how to think about the problem is the same. And this is what we call computational thinking.

As we mentioned above there are neither teaching nor evaluating methods for selecting the people who could become really good data scientists. In our future research, we plan to create an evaluation system to detect data scientist talents. We believe that the presented method could be a very good base for this aim from the very beginning of the course.

Acknowledgement

Katalin Bubnó was supported by the EFOP-3.6.1-16-2016-00022 project.

References

[1] M. Csernoch: Sprego programming. Spreadsheets in Education, 8:3, 2015, pp. 1-35

[2] B. Halassy: Adatmodellezés. 2000 (In Hungarian)

[3] ICAEW: Spreadsheet Competency Framework. 2016

[4] R. B. Millwood: Review of literature on computational thinking. 2018

[5] S. Papert: Mindstorms. Children, computers and powerful ideas. Basic Books, 1980

[6] G. Pólya: How to solve it. 2nd ed., Doubleday, 1957

[7] K. Takácsné Bubnó and V. Takács: Solving word problems by computer programming. Proceedings of Problem Solving in Mathematics Education 2013 Conference (ProMath), A. Ambrus and É. Vásárhelyi, (eds.), Budapest, 2014, pp. 193-208

[8] J. Walkenbach: Excel 2010 Bible. Indianapolis: Wiley, 2010

[9] J. M. Wing: Computational thinking. Communications of the ACM, 49:3, 2006, pp. 33-35

[10] K. Bubnó and V. Takács: The mathability of word problems as initial computer programming exercises. Proceedings of the 8th IEEE International Conference on Cognitive Infocommunications (CoginfoCom), IEEE, 2017, pp. 39-44

[11] P. Biró and M. Csernoch: The mathability of spreadsheet tools. 2015 Proceedings of the 6th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), IEEE, 2015, pp. 105-110

[12] P. Biró and M. Csernoch: Deep and surface structural metacognitive abilities of the first year students of Informatics. 2013 IEEE 4th International Conference on Cognitive Infocommunications (CogInfoCom), IEEE, 2013, pp. 521-526

[13] M. Csernoch, P. Biró, J. Máth and K. Abari: Testing algorithmic skills in traditional and non-traditional programming environments. Inf. Educ. 14, 2015, pp. 175-197

[14] G. Csapó: Sprego virtual collaboration space. 2017 IEEE 8th International Conference on Cognitive Infocommunications (CogInfoCom), IEEE, 2017, pp. 137-142

[15] Baranyi P and Csapo A., Definition and synergies of cognitive infocommunication, Acta Polytechnica Hungarica, 9:1, 2012, pp. 67-83

[16] P. Baranyi and A. Gilányi: Mathability: Emulating and enhancing human mathematical capabilities. Proceedings of the 2013 IEEE 4th International Conference on Cognitive Infocommunications (CogInfoCom), IEEE, 2013, pp. 555-558

[17] K. Chmielewska, A. Gilányi and A. Łukasiewicz: Mathability and mathematical cognition. Proceedings of the 2016 7th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), IEEE, 2016, pp. 245-250

[18] K. Chmielewska and A. Gilányi: Mathability and computer aided mathematical education. 2015 6th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), IEEE, 2015, pp. 473-477

[19] K. Chmielewska and A. Gilányi: Educational context of mathability. Acta Polytechnica Hungarica, 15:5, 2018, pp. 223-237

[20] K. Chmielewska, D. Matuszak: Mathability and Coaching. 8th IEEE Conference on Cognitive Infocommunications (CogInfoCom), IEEE, 2017, pp. 427-431

[21] A. Gilányi, N. Merentes and R. Quintero: Mathability and an animation related to a convex-like property. 7th IEEE Conference on Cognitive Infocommunications (CogInfoCom), IEEE, 2016, pp. 227-232

[22] A. Gilányi, N. Merentes and R. Quintero: Presentation of an animation of the m-convex hull of sets. 7th IEEE Conference on Cognitive Infocommunications (CogInfoCom), IEEE, 2016, pp. 307-308

[23] G. Gy. Borus and A. Gilányi: On a computer program for solving systems of functional equations. 4th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), IEEE, 2013, p. 939

[24] G. Gy. Borus and A. Gilányi: Solving systems of linear functional equations with computer. 4th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), IEEE, 2013, pp. 559-562

[25] G. Gy. Borus and A. Gilányi: Computer assisted solution of systems of two variable linear functional equations. Aequationes Math. 94, 2020, pp. 723-736

[26] K. Bubnó and V. Takács: Cognitive aspects of mathematics-aided computer science teaching. Acta Polytechnica Hungarica, 16:6, 2019, pp. 73-93

[27] A. Ambrus: Some cognitive psychological issues in mathematics education. Gradus 2, 2015, pp. 63-73 (In Hungarian)

[28] A. Kővári and M. Rajcsányi-Molnár: Mathability and creative problem solving in the MaTech Math Competition. Acta Polytechnica Hungarica, 17:2, 2020, pp. 147-161

[29] A. Kővári: Study of algorithm problem-solving and executive function. Acta Polytechnica Hungarica, to appear, 2020

[30] N. Yusupova, G. Shakhmametova and R. Zulkarneev: Complex analysis of medical data with data mining usage. Acta Polytechnica Hungarica, 17:8, 2020, pp. 75-93

[31] G. Kulikov, V. Antonov, A. Fahrullina, L. Rodionova, A. Abdulnagimov and M. Shilina: Formal method of structural-logical identification of functional model of subject area polycubic data matrix. Acta Polytechnica Hungarica, 17:8, 2020, pp. 41-59

Mathability in the Fields of 3D Printing and Modeling

Marianna Zichar

Faculty of Informatics, University of Debrecen, Egyetem tér 1, 4032 Debrecen, Hungary, zichar.marianna@inf.unideb.hu

Abstract: Each term in the title is relatively new, but all of them emerge more and more frequently in various contexts. News regularly introduces some interesting events from the world of 3D printing (that is strongly related to 3D modeling) and Mathability also has an impact on many scientific research projects. This paper makes an attempt to explore how Mathability applies in the field of 3D modeling, by examining the features of Tinkercad. The investigation is completed by analyzing assignments submitted by students. My objective was to get some impression of the extent, of the students use, of their algorithmic skills, in 3D modeling and to explore whether a non-typical programming interface encourages or discourages the application of their programming skills. The results show that the students who majored in the Computer Science Bachelor Program, had a weak affinity, to demonstrate how they could apply tools of higher Mathability, in Tinkercad and thus, the unusual framework and the task did not sufficiently motivate them.

Keywords: 3D printing; 3D modeling; Tinkercad; Mathability

1 Introduction

The word *mathematics* may sound frightening for most people although human life cannot be imagined without applying mathematical knowledge. Children of a very young age start to get to know the world of numbers and this learning curve does not terminate at least till the end of their education. The skills related to mathematics are applied and developed through many different courses and activities, while they are essential to be educated in other fields such as physics, and chemistry.

Human ability and the knowledge of something, differ from each other and have always been subjects of research. Some typical questions from this field: in what extent can be a skill developed, or are there differences in the measure of possible development concerning distinct disciplines [1] [2]? Besides, numerous studies examined for example relations between gender and mathematical ability, which have attracted intensive attention because of the well-known stereotypes about the possible differences [3] [4]. Some typical examples for stereotypes mentioned frequently: women cannot be talented at mathematics, men are better engineers, to be a pilot is not for women [5]. Although society is struggling to demolish these beliefs, they have impacts on many decisions even if they cause some damage.

But disciplines have been evolving (which is a must since they have to reflect on new achievements) and new terms, aspects, points of view appear making it possible to consider new characteristics and starting novel examinations. A similar process happened in 2012 when a new field of science called *cognitive infocommunication* (CogInfoCom) was born. According to the authors it "explores the link between the research areas of infocommunications and cognitive sciences, as well as the various engineering applications which have emerged as a synergic combination of these sciences" [6]. The scientific community justified the raison d'être and more and more scholars have extended their works with examining cognitive aspects of their results as well [7-9]. The interdisciplinary characteristics of this special field are demonstrated with a great number of research works dealing with rather interesting topics, such as cognitive data visualization [10-14], cognitive aspects of human-machine interface systems [15-19], virtual and augmented reality [20-22], or even advanced manufacturing technology [23] [24]. This increased interest was a good reason for giving birth to a comprehensive overview of cognitive infocommunications [25].

The concept of traditional Mathability was redefined and extended in the context of cognitive infocommunication with the objective of investigating any combination of artificial and natural cognitive capabilities relevant to mathematics [26]. The vision of the authors is that infocommunication devices will be characterized by levels of mathematical capabilities. Thus, not only classical physical properties of these devices will be overviewed before selecting the appropriate one for a specific purpose, but this newly defined measure will be taken into consideration too. We have to notice that some properties (mainly related to the hardware) could provide some hints about the mathematical capabilities of devices earlier, but this new, more general measurement can be used to get much more accurate information about it. A device without a driver or software products running on it is nothing else than a pure piece of hardware. Thus, Mathability depends also on the software that has the device run. There are specific programs designed to solve directly mathematical problems, but some of them do not occur isolated but are embedded into a more complex system. Examples for the first case are described in [26], while the other interpretations of Mathability are discussed in numerous papers presented in the next section. The levels of Mathability are different of course, but in each case investigation on Mathability provides an added value to get a complex characterization of the mathematical capability of the device.

These days, the world around us changes dynamically and this change influences the job market. The news tells us that robots and artificial intelligence applications are going to have an increasingly important role, new job types are being given birth while the way of employment is also transforming: more people work from home, or take part-time jobs or become freelancers. In this fast-changing world, the role of knowledge is appreciated and research works dealing with facilitating knowledge transfer are also set in focus.

This paper first, provides an overview of what we already know about Mathability, then, extends the investigation, to the new field of 3D modeling. 3D modeling has not yet been analyzed from the point of view of Mathability. Finally, results of an experiment are presented, where university students enrolled in a course of 3D printing and modeling were involved to provide information on how much they are ready to host greater Mathability levels.

2 Diversity in Research Works Related to Mathability

As mentioned earlier the scientific community has shown a great interest in hosting the redefined concept of Mathability, which is proved by the significant number of papers touching or focusing on Mathability. This section provides a classified overview about a relevant part of published works to understand more clearly the state-of-the-art in this field. Classification is based on the main objectives of investigations that are all related to mathematics to some extent. First, the strictly mathematics-oriented works are presented, then followed by the educational aspects of Mathability.

2.1 Mathematics Measured in Term of Mathability

Solving mathematical problems is often supported with computer-based applications, in which Mathability can be investigated. The complexity of mathematical problems may vary on a large scale thus; the capability of applications can also be diverse.

Solving functional equations and inequalities does not belong to the simplest problems in mathematics that is why several computer algebra systems provide help in dealing with them. A small team of researchers who used Maple for solving problems in the above-mentioned field showed examples of how programs developed in Maple are capable of computing solutions for some classes of functional equations. They demonstrated that the programs increased the level of Mathability of the computer algebra system Maple [27].

Another research work demonstrates an animation developed in the Wolfram Mathematica System to visualize the m-convex hull of sets consisting of finitely many points on the Cartesian plane. The computer-assisted method uses the features of the host system, but finally, it is the intellectual product of the authors considering that they developed a specific program for the animation that is an added value comparing to the original functionality of the system [28].

2.2 Education

Many scholars who are working in educational institutions share their time between research and education. Thus, educational aspects of new theories, concepts, and procedures are often explored and analyzed too. It is trivial that education needs special attention from the scientific community, otherwise we are going to lack the new generation of scholars. Several papers have analyzed the general relationship between education and Mathability [29-31]. This section gives an overview of some distinct fields of education where Mathability has already been investigated.

2.2.1 General Aspects

Before discussing the different fields in education, where Mathability can be interpreted, a more general approach has to be mentioned. Some researchers found that mathabililty could be considered general in terms of computer problem-solving. First, SOLO (Structure of the Observed Learning Outcome) categories of understanding were matched to their typology consisting of approaches used to solve a problem with a computer. Finally, levels of Mathability were assigned to the items of their typology. Although both typologies have 5 categories, matching resulted in only four pairs [32].

2.2.2 Spreadsheet

A study put the spreadsheet tools in the focus of Mathability examinations since the authors found that spreadsheet programs can be used on any level of Mathability [33]. Besides determining the Mathability level of the help surface, click-on usage, or course books, the researchers introduced Sprego that is a concept- and CAAD-based approach to problem-solving in spreadsheet management. Several examples demonstrate in the paper how programming skills can be used when following Sprego methodology, and to what extent algorithmic skills can be developed. To sum up, the appropriate application of Sprego can increase the Mathability level of spreadsheet tools, which finally improves the effectiveness of the application.

2.2.3 Programming

Programming and mathematics are in a close relationship. Thus, it is not surprising that some scientists made research on solving word problems by computer programming. They pointed out that there were certain analogies between classical mathematical problem-solving methods and the steps of thinking when implementing an algorithm for a specific problem. They not only introduce their teaching method but also analyze it from the point of view of Mathability [34].

2.2.4 Mathematics

The relationship between Mathability and mathematics is of high interest not only at the level of mathematicians but also at the level of teaching mathematics. As it was mentioned in the introduction, acquiring knowledge of mathematics is essential for everybody. The depth of the necessary amount of knowledge is different of course, but looking for effective methodologies is always in the focus. Several papers discuss questions related to the quantification of artificial mathematical capabilities and education [27] [35]. Authors suggest that an exact quantification of Mathability of devices, such as calculators, smartphones would be a great help when we have to decide which device is allowed to be used in special situations like an exam. In the study [35], some examples demonstrate how the application of devices with high-level Mathability can support the work of students with difficulties in understanding higher mathematics.

Another arena of mathematics studies is participating in competitions. There is a competition called MaTech where the main goal is to measure the creativity, problem-solving skills, efficiency of teamwork, and application of digital knowledge while solving real mathematical problems. Authors of paper [36] as organizers of this competition present an analysis that identifies the characteristics of the mathematical knowledge and the relationship with the skill of creative presentation and performance.

3 Mathability in 3D Design

Activities in general, including studying, working, traveling, cooking, etc. require an effective collaboration of several skills and knowledge we have or should have. Our success depends on how we combine our different skills and knowledge to find the solution for a problem and to do the necessary steps. Our capabilities are not distinct from each other, they interact and help each other via knowledge transfer.

The SOLO categories of understanding were defined for programming, however this typology is more general and can be extended to any computer related-activities [32]. Three-dimensional modeling is a challenging field of producing models that can serve various purposes. Animations, virtualizations, and engineering use 3D models intensively, and new emerging technologies such as 3D printing also need those [37]. The unique properties of 3D printed objects attract several scientists to research how these exceptional characteristics could be beneficial for their disciplines. But first, an appropriate model has to be designed with 3D modeling software, which process can be and should be introduced into education at several levels [38].

The variegation of a 3D modeling software called Tinkercad motivated me to investigate its Mathability. The following sections introduce and analyze its several interfaces that are followed by discussing an experiment about the willingness of students to adopt higher-level Mathability tools.

3.1 A Scalable 3D Modeling Software -Tinkercad

Tinkercad is a popular, easy-to-use 3D CAD design tool from Autodesk. The primary goal of designing in Tinkercad is to prepare a 3D model for 3D printing. Fortunately, the availability of the simplest, inexpensive desktop FDM (Fused Deposition Modeling) 3D printers, has increased significantly, thus the challenge of making a design has become more attractive for many people. Numerous educational institutions have introduced 3D printing and modeling into their teaching programs, although their exact forms can be diverse. At the same time, we can find pioneers among educators from almost all disciplines who try to benefit from the advantages 3D printing can provide for them [39-42].

There are several reasons why the majority of users who want to learn 3D modeling, select Tinkercad as the first tool to study, these reasons are:

- **Platform**
  As one of the most accessible 3D modeling software around, this online CAD tool can be accessed through your browser giving maximum flexibility in the field of platforms. Your private cloud stores your models that also can be shared easily.
- **Free**
  Tinkercad was designed for mainly educational purposes, thus it was essential to make it free. Educational institutions and people usually try to avoid using proprietary software.
- **Scalability**
  Although the first impression after entering your Tinkercad account is that mainly the young generation will enjoy it, that is not true. Applying the principle of CSM (Constructive Solid Modeling) is just one approach

designing is based on. The possibility of coding at different levels is also available providing the opportunity of matching the tool to the proficiency of the user.

- **Users**
  Scalability implies that a wide range of people can use Tinkercad. While children will operate with dragging and dropping primitives onto the workplane, shape generators already offer the option of sophisticated parametrization, and coding provides unlimited possibilities for users with enough mathematical knowledge.
- **Community**
  A large, vivid, and supportive online community provides help, advice, and plenty of educational resources for users. A design can be shared easily with other users without the need of downloading the model, since models are stored in the cloud.

At different rankings, Tinkercad has always had a good position. According to the last survey of All3DP (leading 3D printing magazine), Tinkercad is the first on the Top 10 Free 3D Modeling Software for Beginners published in January 2020 [43]. Although this list ranked design software for only beginners, Tinkercad can provide an appropriate interface for users from novice till professional and can be integrated into the education easily [44-47].

3.2 Mathability Levels in Tinkercad

Mathability of a device or software communicates information about their capability related to mathematics. The usage of many software can have two different forms. The simplest one relies only on built-in tools, while the other form is about to create something new, such as scripts, codes, functions, which finally enhance the original capabilities of the system, and demands deeper knowledge, and more creativity. Three-dimensional modeling requires many computations, thus investigating Mathability of such software is reasonable.

3.2.1 Built-in Tools

As Figure 1 shows, the first impression about Tinkercad, is that it supports the CSM by providing several predefined primitives that form groups, such as, Basic Shapes, Text and numbers, Characters and Connectors. Unfortunately, these primitives can be combined by applying only two operations: union and difference. To determine the intersection of two primitives (that is a frequently used operator) requires following several steps. The simplest design process, based on building with primitives and operators, is of a low-Mathability.

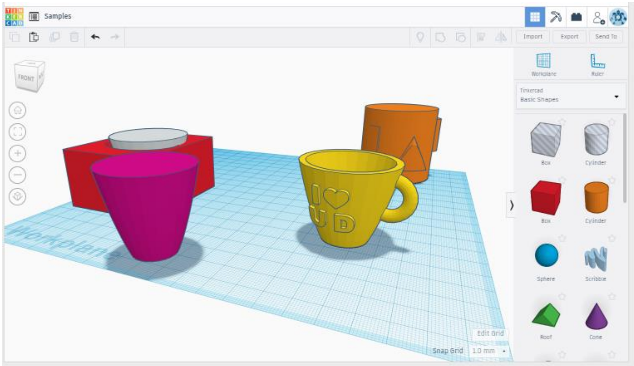


Figure 1
General view of Tinkercad's user interface

Although even basic shapes can be controlled by some parameters, more complex, still built-in tools are available in Tinkercad. They are listed under the group of Shape Generator, and typically provide more sophisticated parameters, and an interactive interface to change their default appearance. For example, Figure 2 represents some settings of a circular array, where the shape of the items arranged around a circle is determined by a sketch. In this case, four points as knots and endpoints of their tangents can be used to influence the shape of items. These tools are still embedded, but their conscious usage requires already more mathematical knowledge, thus its level of Mathability is greater.

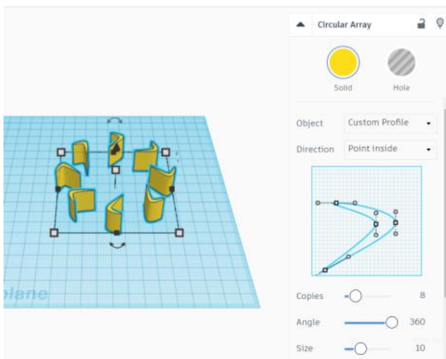


Figure 2
A shape generator in use

3.2.2 Codeblocks

Tinkercad Codeblocks, released in 2018, allows for procedurally generated shapes, using visual programming. To design things in Codeblocks, you simply snap blocks of code together to form a series of sequential commands or actions.

These actions control the shape of your model and result in a 3D design that appears in the 3D viewer. Using Codeblocks can save time, especially when repeating actions form a model, or when manual shape creation, could take an excessive amount of time.

There are six categories of blocks:

- **Shapes**
  Currently, 17 different 3D primitives (such as box, cylinder, sphere, cone, text, etc.) are available that have some parameters to control their exact shape. Every shape is loaded at the origin point at the center of the workspace.
- **Modify**
  This set of blocks contains basic transformations (such as move, scale, rotate), and also blocks to define an object (an individual item of the design identified by a name), to work with an object (such as copy, delete, select all), and to make a grouping.
- **Control**
  Two loop types are available (repeat and count controlled), and one more block for making a pause.
- **Math**
  These blocks can be used to define a variable, to set its value, to generate random numbers, to specify a coordinate triplet, to use basic mathematical functions (such as trigonometrical, logarithmic, rounding, etc.)
- **Data**
  User-defined local variables are listed in this block in order to make it possible to refer them in the blocks.
- **Markup**
  It contains only two blocks: making comments and text outputs.

After having the first impression of the blocks, it is clear that this computational design workspace provides a special interface to automate building from primitives and can save human effort when building a model consisting of many items. Visual programming is attractive for young generations, but people with programming skills can have fun with it and create complex models easily (Figure 3). Designing with Codeblocks can motivate students toward STEAM (Science, Technology, Engineering, Art, Math) studies and can definitely improve their algorithmic skills, which can be rather beneficial for them, by transferring their knowledge to other fields.

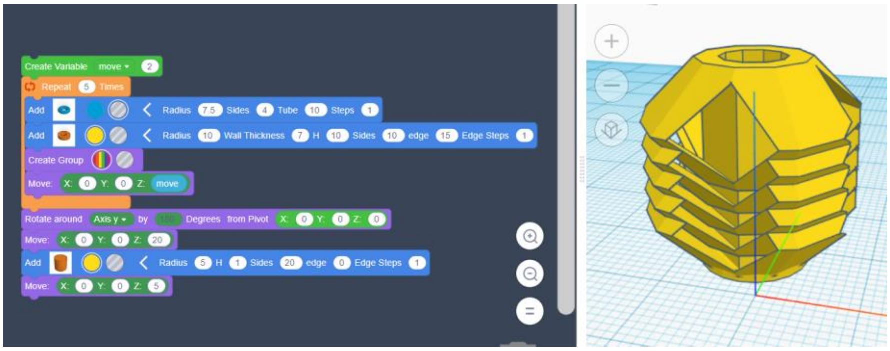


Figure 3
A simple design in Codeblocks

Concerning Mathability of Codeblocks, it has to be noted, that it has already a higher level of Mathability, than the classical workspace introduced in the previous section. Nevertheless, the opportunity for applying variables also contributes to increasing the level of Mathability, since designing with computations requires a more complex way of thinking. Usage of parameters in the control drawer (that can be expanded or collapsed by clicking the arrow) supports minimizing the number of available shapes and introduces a generalization in determining the real form. This can be embarrassing if we consider that the same shape block can result in significantly different outlines depending on the parameter values [48]. Figure 4 represents two pieces of torus differing from each other in only one parameter: the number of sides.

Figure 4
Two different parametrizations of a torus

3.2.3 JavaScript

Although Tinkercad provides many generic geometric shapes that can be combined in many ways to create more complex models, you may need other shapes. The set of shapes can be extended by defining new ones that are created on the creative platform of Tinkercad using JavaScript. If there is a series of shapes you combine frequently, and you describe them with coding in JavaScript then you have created a new shape generator that allows you to automate the creation of the combined model. Already existing shape generators can be customized too, with editing, or also a brand new one can be created from scratch if you have enough knowledge of mathematics and coding.

Figure 4

Empty Shape Generator with the pre-populated code lines

In Tinkercad, this platform has the highest level of Mathability and requires the deepest professional skills from the user. At least a basic understanding of JavaScript is necessary, while the depth of geometry knowledge influences the complexity of a newly developed shape generator. According to the classification in paper [26], Mathability provided by a system can be interpreted in two ways depending on applying only existed tools of the system or enhancing the original set of tools. In the case of writing codes, we face the second approach, while using a shape generator created by anybody else, and shared with us is already belongs to the first class only.

3.3 Experiment with Students

Every semester I have a course called *3D printing and modeling* at the University of Debrecen, where Tinkercad is the first design software taught in classes. The topic of the course is rather attractive for the students and I try to do my best to keep the curriculum interesting and up to date [49]. The spring semester of 2019/20 was the first when I introduced also Codeblocks to the students. The power of control structures, the possibility for parametrization with the help of defining variables were emphasized since these tools provide the main advantages of using Codeblocks. After completing the classes about Codeblocks, I asked them to design something for the sake of demonstrating the capability of Codeblocks, to practice, and to have some fun. All the students were enrolled in Computer Science bachelor degree program, so they were familiar with coding, although they were not enrolled in the same year because this course is optional.

I was expecting stunning models with high complexity, large diversity in applied block types, clear proof of algorithmic skills, and no significant difference between the performance of Hungarian and International Students.

After collecting the URL of models, I have analyzed the codeblocks, and created some statistics to answer some questions about the complexity, number of control statements, degree of diversity. This section demonstrates the most compelling results.

3.3.1 Complexity

The instruction for the assignment was simple: demonstrate the capability of Codeblocks, consider the visual coding as a game, and use the power of loops. Although I thought that the instructions were clear, finally half of the assignments had to be discarded. Unfortunately, some of them ignored the task, while others either copied a sample code or just made a minor modification in a shared code. So, I could examine 15 acceptable assignments.

The first aspect I was interested in examining is the complexity of their codes. Since students are experienced in coding, the tool is not complicated for their programming skills, and the course, is optional. I did expect attractive models. Just for the sake of curiosity, I counted the number of code lines, loops, and shapes generated in loops (Figure 5). It is clear that loops were not preferred, and most students did not want to use programming tools of higher Mathability levels. Only two of them introduced objects that are separated set of codes that could be referenced by their name.

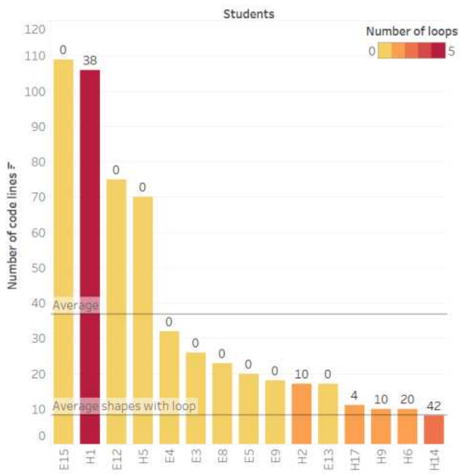


Figure 5
Number of lines, used loops, and shapes generated in loops (number on the bars) by students

3.3.2 Application of their Programming Skills

Three-dimensional designing makes people think of working in a visual environment without writing even one line of code. When introducing the possibility of coding for students majored in computer science you may think, that this way of making 3D models is more comfortable for them. Although the range of classical controlling statements in Codeblocks is limited (two types of loops), other tools (such as variables, work with objects) provide the possibility to use algorithms when building a model. This was my motivation to compare the number of shapes generated in and outside the loops (Figure 6). Concerning the number of shapes coming from a loop, there were two outliers describing two different code structures. The student with the highest number of generated shapes had no shapes outside the loop, while the student in the second place used the second largest number of shapes without loops. Unfortunately, students were not interested in applying their general programming skills, which was surprising.

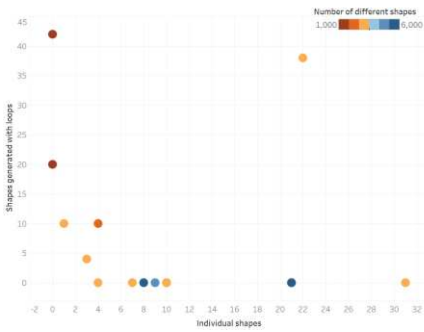


Figure 6

The number of shapes generated in and outside the loops

3.3.3 Variety of Design

Currently, the number of the available shapes in Codeblocks is 17 that is quite impressive. Nevertheless, the number of visually really different shapes is even higher, because the flexible parametrization can result in a significant change in the geometry of the shape as you could see in Figure 4. The students used 13 shapes out of the 17 in their works, but the average number of the different shapes per student was only 3.26 (Figure 7). This shows that the variety of shapes did not motivate them to use more shapes.

We can also consider the Figure 5 and 7 together to look for additional findings. The students who preferred using different shapes and functions did not insert any loops into their codes. The longest codes contained only the average number of different functions and shapes. The model with the highest number of shapes coming from loops consisted of the smallest number of code lines and used only one type of shape and three different function types. The work of student H1 was remarkable, since it was the second longest code, with a high number of shapes generated in several loops, however, the number of different functions and shapes was average.

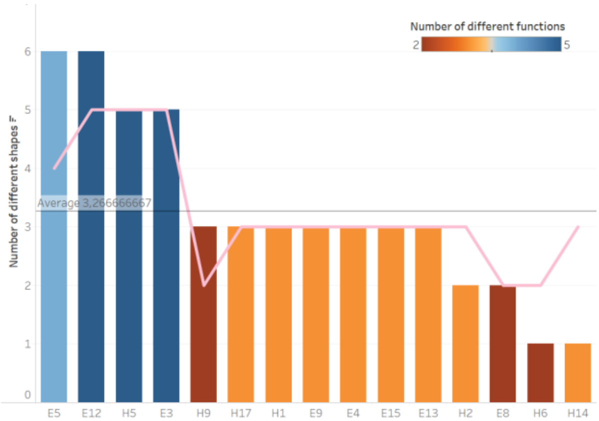


Figure 7
The number of different shapes colored with number of different functions

Figure 8 shows the total number of students per using different shape types. Three shapes (cylinder, box, and text) gave the half portion of the used shapes, which demonstrates again that the students were not engaged in diversity. They preferred using the already known shapes, with different parametrization.

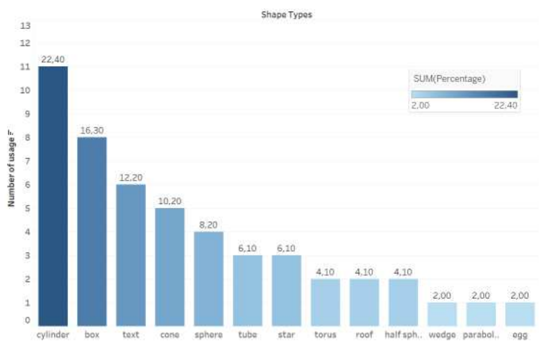


Figure 8
The number students selected the different shapes, and their contributions to the whole

Due to the small number of evaluable codes, I have not analyzed the difference between the performance of Hungarian and International students. Nevertheless, I have to remark here, at least one phenomenon: International students did not use loops or variables at all.

Conclusions

After a thorough Literature Review of the current research related to the concept of Mathability, this paper explored the relationship between Mathability and the emerging technology of, 3D printing and modeling. The concept and the levels of Mathability can also be interpreted in the context of 3D modeling, which was proven by examining the scalability of the popular modeling software product, Tinkercad. It is clear that the concept interpretation is wide, and several levels can be identified. Related to the levels of Mathability, an experiment was carried out, where assignments of university students, majoring in computer science, were analyzed to check their affinity for applying tools of a higher Mathability in Tinkercad. The results showed that my instructions and suggestions were not motivating enough as the students preferred keeping the models very simple and avoided using their programming skills. It has to be mentioned, that the students used the graphical designing interface of Tinkercad, for four weeks, which could have a negative impact on their willingness to use coding, even if Codeblock generation has a user-friendly interface. At the same time, some of them felt that use of Codeblocks was not designed for users with programming skills, which is not true. Complex models cannot be designed in Tinkercad by only using the traditional interface.

I plan to repeat the experiment in the upcoming years, with more clear and probably, more effective instructions. In the case of a higher number of evaluable codes, more aspects should be analyzed. Also, a follow-up questionnaire could provide more information concerning the reasons for a students' activity or inactivity in coding. Data visualization could be enhanced by online interactive dashboards, that would facilitate the identification of more complex relationships among the data.

Acknowledgement

This work was supported by the construction EFOP-3.6.3-VEKOP-16-2017-00002. The project was supported by the European Union, co-financed by the European Social Fund.

References

[1] B. Göranzon, I. Josefson (Eds.): Knowledge, skill and artificial intelligence, Springer Science & Business Media, 2012

[2] O. Uzumcu, E. Bay: The effect of computational thinking skill program design developed according to interest driven creator theory on prospective teachers. Education and Information Technologies, Vol. 26, 2021, pp. 565-583

[3] S. A. Sorby, C. Leopold, R. A. Górska: Cross-Cultural Comparisons of Gender Differences in the Spatial Skills of Engineering Students, Journal of Women and Minorities in Science and Engineering, Vol. 5, No. 3, 1999, pp. 279-281

[4] B. Nemeth, M. Hoffmann: Gender differences in spatial visualization among engineering students, Annales Mathematicae et Informaticae Vol. 33, 2006, pp. 169-174

[5] A. Powell, A. Dainty, B. Bagilhole: Gender stereotypes among women engineering and technology students in the UK: lessons from career choice narratives. European Journal of Engineering Education, 37(6), 2012, pp. 541-556

[6] P. Baranyi, Á. Csapó: Definition and Synergies of Cognitive Infocommunications, Acta Polytechnica Hungarica, 9(1), 2012, pp. 67-83

[7] I. Papp, R. Tornai, M. Zichar: What 3D technologies can bring into the education: The impacts of acquiring a 3D printer, Proceedings of 7th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Wroclaw, 2016, pp. 257-261

[8] P. T. Kovács, T. B. Holografika: 3D Display technologies and effects on the human vision system, Proceedings of 2nd IEEE Conference on Cognitive Infocommunications (CogInfoCom), Budapest, 2011, pp. 1-33

[9] Sz. Szeghalmy, M. Zichar, A. Fazekas: Gesture-based computer mouse using Kinect sensor, Proceedings of 5th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Vietri sul Mare, 2014, pp. 419-424

[10] Gy. Papp, I. Papp, R. Kunkli: Three-dimensional connection visualization based on tabular data, Proceedings of 8th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Debrecen, 2017, pp. 289-290

[11] R. Idzikowski, K. Kluwak, T. Nowobilski, T. Zamojski: Analysis of possibility of visualization of danger factors in the building environment, Proceedings of 8th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Debrecen, 2017, pp. 363-368

[12] Á. Török, Zs. Gy. Török, B. Tölgyesi: Cluttered centres: interaction between eccentricity and clutter in attracting visual attention of readers of a 16th century map, Proceedings of 8th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Debrecen, 2017, pp. 433-438

[13] D. Balla, T. Mester, Á. Botos, T. J. Novák, M. Zichar, J. Rásó, A. Karika: Possibilities of spatial data visualization with web technologies for cognitive interpretation, Proceedings of 8th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Debrecen, 2017, pp. 17-20

[14] D. Sik, K. Csorba, P. Ekler: Implementation of a Geographic Information System with Big Data Environment on Common Data Model, Proceedings of 8th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Debrecen, 2017, pp. 181-184

[15] A. Kővári, J. Katona, C. Costescu: Evaluation of Eye-Movement Metrics in a Software Debugging Task using GP3 Eye Tracker, Acta Polytechnica Hungarica, 17(2), 2020, pp. 57-76

[16] A. Kővári, J. Katona, C. Costescu (2020): Quantitative Analysis of Relationship Between Visual Attention and Eye-Hand Coordination, Acta Polytechnica Hungarica, Vol. 17, No. 2, 2020, pp. 77-95

[17] Á. Török: From human-computer interaction to cognitive infocommunications: a cognitive science perspective, Proceeding of 7th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Wroclaw, 2016, pp. 433-438

[18] A. Fazekas, K. Bertók: Face recognition on mobile platforms, Proceedings of 7th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Wroclaw, 2016, pp. 37-42

[19] Sz. Szeghalmy, M. Zichar, A. Fazekas: Gesture-based computer mouse using Kinect sensor, Proceedings of 5th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Vietri sul Mare, 2014, pp. 419-424

[20] I. Horváth: Innovative engineering education in the cooperative VR environment, Proceedings of 7th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Wrocław, 2016, pp. 359-364

[21] Gy. Bujdosó, O. C. Novac, T. Szimkovics: Developing cognitive processes for improving inventive thinking in system development using a collaborative virtual reality system, Proceedings of 8th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Debrecen, 2017, 79-84

[22] M. Csernoch, Gy. Bujdosó, M. Borbély, E. Dani, M. Némethi-Takács, K. Koltay, L. Balázs: LibSearchNet: Analyses of library log files to identify search flows, Proceedings of 4th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Budapest, 2013, pp. 543-548

[23] F. Erdős, R. Németh: AMT-based Real-Time, Inter-Cognitive Communication Model, Acta Polytechnica Hungarica, 16(6), 2019, pp. 115-127

[24] I. Papp, M. Zichar: 3D Modeling and Printing Interpreted in Terms of Cognitive Infocommunication In: Klempous, Ryszard; Nikodem, Jan; Zoltán Baranyi, Péter (ed.) Cognitive Infocommunications, Theory and Applications Cham (Switzerland), Switzerland: Springer International Publishing, 2019, pp. 365-389

[25] P. Baranyi, Á. Csapó, Gy. Sallai: Cognitive Infocommunications (CogInfoCom), Springer, 2015

[26] P. Baranyi, A. Gilányi: Mathability: emulating and enhancing human mathematical capabilities, in Proceedings of 4th IEEE Conference on

Cognitive Infocommunications (CogInfoCom), Budapest, 2013, pp. 555-558

[27] G. Gy. Borus, A. Gilányi: Solving systems of linear functional equations with computer, Proceedings of 4th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Budapest, 2013, pp. 559-562

[28] A. Gilányi, N. Merentes, R. Quintero: Mathability and an animation related to a convex-like property, Proceedings of 7th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Wrocław, 2016, pp. 227-232

[29] K. Chmielewska, A. Gilányi: Educational context of Mathability, Acta Polytechnica Hungarica, 15, 2019, pp. 223-237

[30] K. Chmielewska, A. Gilányi, A. Łukasiewicz: Mathability and Mathematical Cognition, Proceedings of 7th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Wrocław, 2016, pp. 245-250

[31] K. Chmielewska, D. Matuszak: Mathability and coaching, Proceedings of 8th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Debrecen, 2017, pp. 427-432

[32] P. Biró, M. Csernoch: The Mathability of computer problem solving approaches, Proceedings of 6th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Győr, 2015, pp. 111-114

[33] P. Biró, M. Csernoch: The Mathability of spreadsheet tools, Proceedings of 6th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Győr, 2015, pp. 105-110

[34] K. Bubnó, V. L. Takács: Cognitive Aspects of Mathematics-aided Computer Science Teaching, Acta Polytechnica Hungarica, Vol. 16, No. 6, 2019, pp. 73-93

[35] K. Chmielewska, A. Gilányi: Mathability and computer aided mathematical education, Proceedings of 6th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Győr, 2015, pp. 473-477

[36] A. Kovari, M. Rajcsanyi-Molnar: Mathability and Creative Problem Solving in the MaTech Math Competition, Acta Polytechnica Hungarica, Vol. 17, No. 2, 2020, pp. 147-161

[37] F. Erdős, R. Nemeth: Inter-cognitive Communication Model Using Additive Manufacturing Technology, Proceedings of 9th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Budapest, 2018, pp. 217-222

[38] I. Papp, M. Zichar: Modeling and printing in 3D at higher education, Proceedings of 10th International Conference on Applied Informatics, Eger, 2017, pp. 235-243

[39] S. Ford, T. Minshall, Invited review article: Where and how 3D printing is used in teaching and education, Additive Manufacturing 25, 2019, pp. 131-150

[40] M. Zichar: Added values of additive manufacturing to geoinformatics Proceedings of VIII. Theory meets practice in GIS, Debrecen, 2017, pp. 437-443

[41] I. M. Alhamad, K. A. Waleed, H. Z. Ali, H. AlJassmi: 3D Printing Applications in Mechanical Engineering Education, In: Khine, Myint Swe; Ali, Nagla (Eds.) Integrating 3D Printing into Teaching and Learning, Brill Sense, 2020, pp. 90-131

[42] C. Schelly, G. Anzalone, B. Wijnen, J. M. Pearce: Open-source 3-D printing technologies for education: Bringing additive manufacturing to the classroom, Journal of Visual Languages & Computing, 28, 2015, pp. 226-230

[43] https://all3dp.com/1/best-free-3d-modeling-software-for-beginners/

[44] L. M. Díaz, C. M. Hernández, A. V. Ortiz, L. S. Gaytán-Lugo: Tinkercad and Codeblocks in a Summer Course: an Attempt to Explain Observed Engagement and Enthusiasm, 2019 IEEE Blocks and Beyond Workshop (B&B), Memphis, TN, USA, 2019, pp. 43-47

[45] P. R. Sajil Raj, A. Anshadh, S. B. T. Raj, A. N. Ahsana: Design of an innovative coconut grating machine using Tinkercad, International Journal of Research in Mechanical Engineering, Vol. 4, Issue 3, 2016, pp. 178-182

[46] S. Yi, U. Jung, Y. Lee: A Study on the direction of 3D Modeling Education Considering Computational Thinking Factors at Elementary School in South Korea, Proceedings of E-Learn: World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education, Vancouver, 2017, pp. 505-508

[47] O. Ng: Exploring the use of 3D Computer-Aided Design and 3D Printing for STEAM Learning in Mathematics, Digital Experiences in Mathematics Education 3, 2017, 257-263

[48] M. Zichar, I. Papp: Interaction between 3D printing and geometry studies Proceedings of 18th International Conference on Geometry and Graphics: 40th Anniversary, Milan, 2018, pp. 1177-1190

[49] M. Zichar: Exploring the role and possibilities of 3D printing in IT studies: Is my curriculum good enough?, Proceedings of 10th IEEE Conference on Cognitive Infocommunications (CogInfoCom), 2019, pp. 221-226

On Packing of Unequal Squares in a Rectangle

Antal Joós

University of Dunaújváros, Institute of Computer Engineering, Department of Mathematics, Táncsics M. u. 1/a, 2400 Dunaújváros, Hungary, joosa@uniduna.hu

Abstract: It is known that the sum of the squares of the reciprocals, of positive integer numbers, is finite. It can be asked… What is the smallest area rectangle into which all the squares of sides of length the reciprocals of the positive integers can be packed? In connection with the investigations related to mathability and to applications of computer assisted methods, for considering mathematical problems, an improvement for the best known ϵ *is presented, herein. The GNU program, Octave, was used for the calculations.*

Keywords: Mathability; Cognitive Infocommunications; Computer Assisted Methods; Packing; Square

1 Introduction

Mathability refers to a branch of cognitive infocommunications that investigates any combination of artificial and natural cognitive capabilities, relevant to mathematics, including a wide spectrum of areas ranging from low-level arithmetic operations, to high-level symbolic reasoning. The concept of Cognitive Infocommunications (CogInfoCom) was introduced in the paper [1]. Some of its further general properties were described in the papers [2] and [3] and in the book [4]. The educational aspects of CogInfoCom and mathability were investigated, among others, in [5-12] while other CogInfoCom related applications of cognitive capabilities are presented in [13-20].

Questions related to mathability and to computer based methods for investigations of mathematical problems have been studied by several authors in recent years [21-25]. T work has contributed to these investigations. A computer assisted method for a packing of squares, of sides of length $\frac{1}{2}, \frac{1}{3}, \frac{1}{4}, \ldots$ is presented.

The paper [26], motivated me to present a computer assisted method for a problem of Meir and Moser [27]. The calculation is performed in Octave, available at https://www.gnu.org/software/octave/download.

2 The Problem

It is said the squares $S_1, S_2, S_3, \ldots$ can be packed into a rectangle if it is possible to apply translations and rotations to the sets S_n so that the resulting translated and rotated squares are contained in the rectangle and have mutually disjoint interiors.

Meir and Moser [27], in 1968, originally noted that since:

$$\sum_{i=2}^{\infty} \frac{1}{i^2} = \frac{\pi^2}{6} - 1 \tag{1}$$

it is reasonable to ask whether the set of squares of sides of length $\frac{1}{2}, \frac{1}{3}, \frac{1}{4}, \ldots$, which is called the reciprocal squares, can be packed into a rectangle of area $\frac{\pi^2}{6} - 1$. Failing that, find the smallest ϵ such that the reciprocal squares can be packed into a rectangle R of area $\frac{\pi^2}{6} - 1 + \epsilon$. This question can be found in e.g. [28].

Meir and Moser [27] in 1968 showed that the smallest square containing the reciprocal squares is the square of side $\frac{5}{6}$ which shows that $\epsilon < \frac{1}{205}$. Jennings [29] in 1994 gave a rectangle of dimensions $\frac{47}{60} \times \frac{5}{6}$ containing the reciprocal squares which shows that $\epsilon < \frac{1}{127}$

Ball [30] in 1996 gave a rectangle of dimensions $\frac{629}{1\,000} \times \frac{31}{30}$ containing the reciprocal squares which shows that $\epsilon < \frac{1}{198}$

Paulhus [26] in 1997 gave a rectangle of dimensions:

$$0.5 \times \left(2\left(\frac{\pi^2}{6} - 1\right) + 1.606\,553\,066 \cdot 10^{-9}\right) \tag{2}$$

containing the reciprocal squares which shows that:

$$\epsilon < \frac{1}{1\,244\,918\,662} = 8.032\,653\,301 \ldots \cdot 10^{-10} \tag{3}$$

The author [31] in 2018 has found a mistake in the proof of Paulhus. Grzegorek and Januszewski [32] in 2019 filled this gap in the proof of Paulhus. In this paper a new estimate for ϵ is presented.

3 Construction

Theorem 1

The rectangle of dimensions $0.5 \times \left(2\left(\frac{\pi^2}{6} - 1\right) + 1.363\,813\,307\,2 \cdot 10^{-9}\right)$ contains the reciprocal squares which shows that $\epsilon < 6.819\,066\,536 \cdot 10^{-10}$

From the following construction it comes the proof of the theorem.

The square of size $\frac{1}{i}$ is referred by (the square) i. In this paper the width of a rectangle will always refer to the shorter side and the length will always refer to the longer side of the rectangle. Let R be the rectangle of dimensions $\frac{1}{2}\times\ 2\left(\frac{\pi^2}{6}-1\right)$ in which the squares are packed first. It is assumed, that the width of R is horizontal. Let $l_0 = 0.000\ 019\ 03$. Let R' be the square of side length l_0.

Let $A = 1\ 622\ 971\ 324$, $B = 1\ 648\ 721\ 271$, $C = 2\ 675\ 827\ 341$, $D = 2\ 718\ 281\ 828$, $E = 2\ 761\ 408\ 695$. The numbers A, B, C, D and E comes from [26]. Let $n_0 = E + 1$, $n_i = \lfloor n_{i-1}(1 + l_0)\rfloor$ for $i \geq\ 1$ where $\lfloor . \rfloor$ is the floor function and $n'_{18} = 2\ 762\ 386\ 911$. Observe the squares from n_{i-1} to $n_i - 1$ fit in a row of R'. Let $A_1 = 1\ 016\ 225\ 800$, $A_2 = 1\ 000\ 000\ 440$, $C' = 2\ 633\ 103\ 139$ and $C_0 = 2\ 674\ 879\ 766$, ..., $C_{18} = 2\ 675\ 796\ 170$ so that

$$\sum_{i=C_j}^{C_{j+1}-1}\frac{1}{i} < l_0 \leq \sum_{i=C_j}^{C_{j+1}}\frac{1}{i} \quad \text{for } j = 0, \dots, 18 \tag{4}$$

The numbers $C_1, \dots, C_{18}$ are calculated with the help of Octave and Lemma 1.

Lemma 1. The following is true:

$$\ln\frac{n+m+1}{n} < \frac{1}{n} + \cdots + \frac{1}{n+m} < \ln\frac{n+m}{n-1} \tag{5}$$

where n and m are positive integers and $n \neq 1$

Proof of Lemma 1. After using the lower and upper sums of the function

$x \mapsto \frac{1}{x}$ the estimates are trivial.

By Lemma 1, it is true:

$$\frac{1}{n} + \cdots + \frac{1}{n+m} \approx \frac{\ln\frac{n+m+1}{n} + \ln\frac{n+m}{n-1}}{2} \tag{6}$$

The numerical estimations based on the Lemma 1. The following two short Octave retval functions help the calculations:

```
function retval = distU (k,v)
  retval = log(v/(k-1));
endfunction
function retval = distL (k,v)
  retval = log((v+1)/k);
endfunction.
```

By Lemma 1, the function distL(k,v) returns an lower bound of the sum $\frac{1}{k}+\cdots+\frac{1}{v}$ and the function distU(k,v) returns an upper bound of the sum $\frac{1}{k}+\cdots+\frac{1}{v}$.

The following is used:

Lemma 2. The squares C_0, C_0+1, ..., C, n'_{18}, $n'_{18}+1$, ..., $n_{100001}-1$,

$n_{100\,251}$, $n_{100\,251}+1$, ... can be packed in R'.

Proof of Lemma 2. The squares from C_0 to C are packed in rows of length no greater than l_0 in the square R'.

If $1 \leq i \leq 18$, then in the ith row the squares go from C_{i-1} to C_i-1 (Figure 1). Observe, the squares from C_{i-1} to C_{i-1} fit in the i^{th} row.

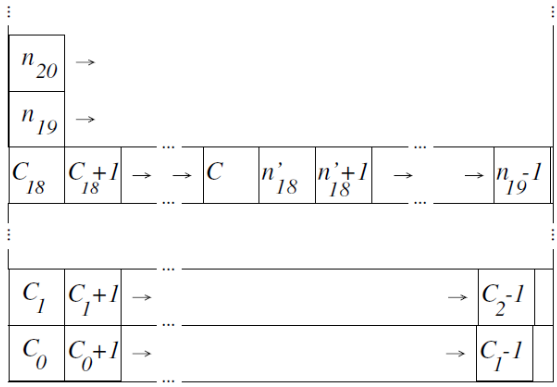


Figure 1

The ith row in R' ($1 \leq i \leq 19$)

By Lemma 1 and Octave, in the 19th row the squares go from C_{18} to C and from n'_{18} to $n_{19}-1$. If $20 \leq i \leq 100\,001$, then in the ith row the squares go from n_{i-1} to n_i-1. Observe, the squares from n_{i-1} to n_i-1 fit in the i^{th} row. If $100\,002 \leq i$, then in the i^{th} row the squares go from n_{i+249} to $n_{i+250}-1$ (Figure 2). Observe, the squares from n_{i+249} to $n_{i+250}-1$ fit in the i^{th} row.

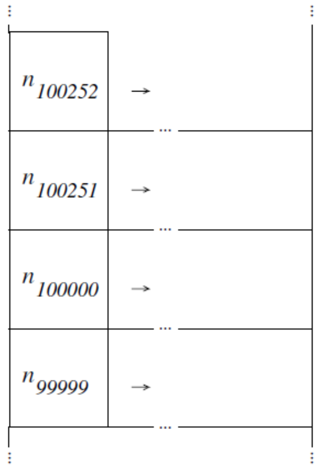


Figure 2
The ith row in R' ($100\,000 \leq i \leq 100\,003$)

Now:

$$n_i = \lfloor n_{i-1}(1+l_0)\rfloor > n_{i-1}\left(1+l_0-\frac{1}{n_{i-1}}\right)$$
$$> n_{i-2}\left(1+l_0-\frac{1}{n_{i-1}}\right)\left(1+l_0-\frac{1}{n_{i-2}}\right)$$
$$> n_{i-2}\left(1+l_0-\frac{1}{n_{i-2}}\right)^2 > \cdots > n_{18}\left(1+l_0-\frac{1}{n_{18}}\right)^{i-18} \tag{7}$$

for $i > 18$, Thus:

$$\sum_{i=1}^{19}\frac{1}{C_{i-1}} + \sum_{i=20}^{100\,001}\frac{1}{n_{i-1}} + \sum_{i=100\,002}^{\infty}\frac{1}{n_{i+249}}$$
$$< 0.000\,000\,007\,101\,90\ldots + \sum_{i=20}^{100\,001}\frac{1}{n_{18}\left(1+\mathrm{l}_0-\frac{1}{\mathrm{n}_{18}}\right)^{i-19}}$$
$$\ldots + \sum_{i=100\,002}^{\infty}\frac{1}{n_{18}\left(1+\mathrm{l}_0-\frac{1}{\mathrm{n}_{18}}\right)^{i+231}}$$
$$= 0.000\,000\,007\,101\,90\ldots + \frac{1}{n_{18}\left(1+\mathrm{l}_0-\frac{1}{\mathrm{n}_{18}}\right)}\sum_{i=0}^{99\,981}\frac{1}{\left(1+\mathrm{l}_0-\frac{1}{\mathrm{n}_{18}}\right)^{i}}$$

$$+\frac{1}{n_{18}\left(1+l_0-\frac{1}{n_{18}}\right)^{100\,233}}\sum_{i=0}^{\infty}\frac{1}{\left(1+l_0-\frac{1}{n_{18}}\right)^{i}}$$

$$= 0.000\,000\,007\,101\,90\ldots+\frac{1}{n_{18}\left(1+l_0-\frac{1}{n_{18}}\right)}\frac{1-\frac{1}{\left(1+l_0-\frac{1}{n_{18}}\right)^{99\,982}}}{1-\frac{1}{1+l_0-\frac{1}{n_{18}}}}$$

$$+\frac{1}{n_{18}\left(1+l_0-\frac{1}{n_{18}}\right)^{100\,233}}\frac{1}{1-\frac{1}{1+l_0-\frac{1}{n_{18}}}}<0.000\,019\,017<l_0 \tag{8}$$

Octave was used to the numerical calculations. Thus the squares fit in R', which is the statement of the lemma.

Proof of Theorem 1. It is assumed that the squares up to 10^9 are packed in R as in [26]. By [26], there is a rectangle R_L of length and width at most l_0, which has no common interior point with the squares up to 10^9. By Lemma 2, it is necessary to find a place to pack the squares from 10^9+1 to C_0-1, from $C+1$ to $n'_{18}-1$ and from $n_{100\,001}$ to $n_{100\,251}-1$.

Let the squares from 10^9+1 to C_0-1 and from $C+1$ to $n'_{18}-1$ be packed into a rectangle R_N of length $\frac{1}{2}$ as in Figure 3. By [26], the squares from 10^9+1 to B and from $C+1$ to D can be arranged as in Figure 3. By Lemma 1, $\frac{1}{C'}+\cdots+\frac{1}{C_0-1}<\frac{1}{A}+\cdots+\frac{1}{B}$.

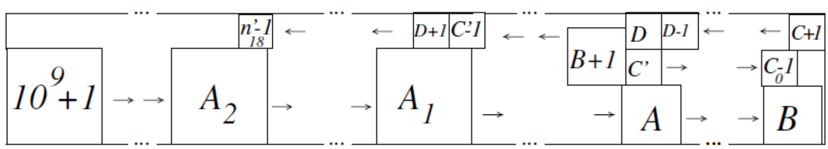


Figure 3

The squares in R_N

It is shown that the highest horizontal edge belongs to the square $C'-1$ thus, the width of the rectangle R_N is $\frac{1}{A_1}+\frac{1}{C'-1}=\ 1.363\,813\,307\,18\ldots\cdot 10^{-9}$.

First, it is shown that the highest horizontal edge belongs to the square $C'-1$ among the squares $B+1$, ..., $C'-1$. The square $C'-y$ *sits* on the square A_1+x if the relative interior of the bottom side of $C'-y$ and the relative interior of the upper side of A_1+x have a nonempty intersection. Since:

$$\frac{\frac{1}{A_1}}{\frac{1}{C'-1}} = 2.5 \ldots \tag{9}$$

at most three squares sit on the square $A_1 + x$ if x is a (small) positive integer. It is assumed, that the square $C' - y$ sits on the square $A_1 + x$. Thus

$$\frac{1}{A_1} + \frac{1}{C'-1} > \frac{1}{A_1 + x} + \frac{1}{C'-1-3x} \geq \frac{1}{A_1 + x} \frac{1}{C'-y} \tag{10}$$

if $0 < x \leq x_1$, where $x_1 = 350\ 300\ 705$ (the value of x_1 comes from the Octave). By Lemma 1, the square $y_1 = 1\ 958\ 123\ 269$ sits on the square $A_1 + x_1$, but $y_1 + 1$ does not sit on $A_1 + x_1$. Since:

$$\frac{\frac{1}{A_1 + x}}{\frac{1}{y_1}} = 1.4 \ldots \tag{11}$$

at most two squares sit on the square $A_1 + x_1 + x$ if x is a (small) positive integer. It is assumed, that the square $y_1 - y$ sits on $A_1 + x_1 + x$. Thus:

$$\frac{1}{A_1} + \frac{1}{C'-1} > \frac{1}{A_1 + x_1 + x} + \frac{1}{y_1 - 2x} \tag{12}$$

$$\geq \frac{1}{A_1 + x_1 + x} + \frac{1}{y_1 - y} \tag{13}$$

if $0 < x \leq x_2$, where $x_2 = 334\ 746\ 954$ (the value of x_2 comes from the Octave). Since $A_1 + x_1 + x_2 > A$, the highest horizontal edge belongs to the square $C' - 1$ among the squares $B + 1$, ..., $C' - 1$.

Similarly:

$$\frac{1}{A_2} + \frac{1}{n'_{18} - 1} > \frac{1}{A_2 + x} + \frac{1}{n'_{18} - 1 - 3x} \tag{14}$$

if $0 < x \leq A_2 - A_1 - 1$

and

$$\frac{1}{C'}+\frac{1}{D'-1}>\frac{1}{C'+x}+\frac{1}{D-2x} \tag{15}$$

if $0<x\leq\ C_0-C'-2$

The candidates of the width of the rectangle R_N are

$$\frac{1}{A_2}+\frac{1}{n'_{18}-1}=1.362\ 005\ 3\ \ldots\cdot 10^{-9}$$

$$\frac{1}{A_1}+\frac{1}{C'-1}=1.363\ 813\ 307\ 19\ \ldots\cdot 10^{-9}$$

$$\frac{1}{A}+\frac{1}{C'}\ +\frac{1}{D-1}=1.363\ 813\ 307\ 18\ \ldots\cdot 10^{-9} \tag{16}$$

Thus the width of the rectangle R_N is $\frac{1}{A_1}+\frac{1}{C'-1}=1.363\ 813\ 307\ 19\ \ldots\cdot 10^{-9}$

It is necessary to find a place to pack the squares $n_{100\ 001}$ to $n_{100\ 251}-1$.

Let:

$$b_1=\frac{1}{n_{18}\left(1+l_0-\frac{1}{n_{18}}\right)^{99\ 983}}\ =5.400\ 354\ 04\ \ldots\cdot\ 10^{-11} \tag{17}$$

Now:

$$\frac{1}{n_i}\leq\frac{1}{n_{100\ 001}}<\frac{1}{n_{18}\left(1+l_0-\frac{1}{n_{18}}\right)^{99\ 983}}=b_1 \tag{18}$$

for $i=100\ 001,\ldots,100\ 250$ and let:

$$b_2=\frac{1}{n_{100\ 001}}+\frac{1}{n_{100\ 001}+1}+\cdots+\frac{1}{n_{100\ 251}-1}<250\cdot\ l_0 \tag{19}$$

The square $y_n=1\ 656\ 583\ 751$ sits on the square $x_n=1\ 615\ 268\ 375$. (Observe, y_n-1 does not sit on x_n.) Thus:

$$\sum_{\mathrm{i=B+1}}^{\mathrm{y_n-1}}\frac{1}{i}\ <250\cdot\ \mathrm{l_0}<\sum_{\mathrm{i=B+1}}^{\mathrm{y_n}}\frac{1}{i} \tag{20}$$

and

$$\sum_{i=x_n+1}^{A-1} \frac{1}{i} < 250 \cdot l_0 < \sum_{i=x_n}^{A-1} \frac{1}{i} \tag{21}$$

Let:

$$h = \frac{1}{A_1} + \frac{1}{C' - 1} - \frac{1}{x_n} - \frac{1}{y_n} = 1.410\,719\,974\,85 \ldots \cdot 10^{-10} \tag{22}$$

Since the highest horizontal edge belongs to the square y_n among the squares $B + 1$, ..., y_n, there is a rectangle R_f of dimensions $250 \cdot l_0 \times h$ which has no common interior point with the squares up to $n'_{18} - 1$. Since $b_1 < h$ and $b_2 < 250 \cdot l_0$, the squares from $n_{100\,001}$ to $n_{100\,251} - 1$ fit in R_f.

Thus the reciprocal squares are contained in a rectangle of dimensions

$$\frac{1}{2} \times \left(2\left(\frac{\pi^2}{6} - 1\right) + 1.363\,813\,307\,19 \ldots \cdot 10^{-9}\right) \tag{23}$$

which shows that $\epsilon \leq 6.819\,066\,535\,97 \ldots \cdot 10^{-10}$

Conclusions

From the above proof, it should be recognised, that performing difficult calculations with the help of computers and/or suitable programs, can be an easy task. Without a computer, the calculations on a piece of paper, indeed, take a very long time.

Calculations with Octave and the two short retval functions, is an easy task. The numbers A, B, C, D and E come from [26]. The numbers $A_1, A_2, C', C_0, C_1, \ldots, C_{18}$ and the width of the rectangle R_N are calculated with the help of Octave. By Lemma 1, the control of round-off errors is achieved.

The packing question in this paper, for ϵ, was asked back in 1968, and the question is still open. In the papers [26] [27] [29] [30] improved estimates for the value of ϵ can be found, but these estimates were not final, as you can see by the evidence of this paper. This short work should inspire authors to closely examine long standing mathematical questions, with the help of a computer.

Acknowledgement

This work was supported by EFOP-3.6.1-16-2016-00003 funds. Establishment of long-term R and D and I processes at the University of Dunaújváros.

References

[1] Baranyi P., Gilányi A.: Mathability: emulating and enhancing human mathematical capabilities, Proceedings of the 4th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Budapest, Hungary, 2013, pp. 555-558

[2] Baranyi P., Csapo A., Varlaki P.: An overview of research trends in coginfocom. Proceedings of the 18th International Conference on Intelligent Engineering Systems (INES), Tihany, Hungary, 2014, pp. 181-186

[3] Török M., Tóth J., Szöllősi A.: Foundations and perspectives of mathability in relation to the coginfocom domain, Proceedings of the 4th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Budapest, Hungary, 2013, pp. 869-872

[4] Baranyi P., Csapo A., Sallai G.: *Cognitive Infocommunications (CogInfoCom)*, Springer, 2015

[5] Biró P., Csernoch M.: Deep and surface metacognitive processes in non-traditional programming tasks, Proceedings of the 5th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Vietri sul Mare, Italy, 2014, pp. 49-54

[6] Biró P., Csernoch M.: The mathability of computer problem solving approaches, Proceedings of the 6th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Győr, Hungary, 2015, pp. 111-114

[7] Chmielewska K., Ciskowska W., Glazik D, Marcinek D., Wojciechowska K., Gilányi A.: Learnability – are we ready for distance learning?, Proceedings of the 11th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Online, 2020, pp. 465-470

[8] Chmielwska K., Gilányi A., Łukasiewicz A.: Mathability and mathematical cognition, Proceedings of the 7th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Wroclaw, Poland, 2016, pp. 245-250

[9] Chmielwska K., Gilányi A., Mathability and computer aided mathematical education, Proceedings of the 6th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Győr, Hungary, 2015, pp. 473-477

[10] Chmielewska K., Gilanyi A.: Educational context of mathability, *Acta Polytechnica Hungarica*, Vol. 15, No. 5, 2018, pp. 223-237

[11] Kővári A., Rajcsányi-Molnár M.: Mathability and Creative Problem Solving in the MaTech Math Competition, *Acta Polytechnica Hungarica*, 17(2), 2020, pp.147-161

[12] Szi B., Csapo A.: An outline of some human factors contributing to mathability research, Proceedings of the 5th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Vietri sul Mare, Italy, 2014, pp. 583-586

[13] Bognár L., Fauszt T.: Different learning predictors and their effects for Moodle Machine Learning models, 2020 11th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Mariehamn, Finland, 2020, pp. 405-410

[14] Bognár L., Fauszt T., Nagy G. Zs.: Analysis of Conditions for Reliable Predictions by Moodle Machine Learning Models, to International Journal of Emerging Technologies in Learning, accepted

[15] Bognár L., Fauszt T.: The Effectiveness of Different Sets of Learning Predictors in Moodle Machine Learning Models, *Acta Polytechnica Hungarica*, accepted

[16] Katona J. Kovari A, Heldal I., Costescu C., Rosan A., Demeter R., Thill S., Stefanut T.: Using Eye- Tracking to Examine Query Syntax and Method Syntax Comprehension in LINQ, Proceedings of the 12th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Naples, Italy, 2020, pp. 437-444

[17] Katona J., Ujbanyi T., Sziladi G., Kovari A.: Examine the effect of different web-based media on human brain waves, Proceedings of the 8th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Debrecen, Hungary, 2017, pp. 407-412

[18] Katona J., Ujbanyi T., Sziladi G., Kovari A.: Electroencephalogram-Based Brain-Computer Interface for Internet of Robotic Things, *In Cognitive Infocommunications, Theory and Applications*, 2019, pp. 253–275

[19] Kővári A., Katona, J., Costescu, C.: Evaluation of Eye-Movement Metrics in a Software Debugging Task using GP3 Eye Tracker, *Acta Polytechnica Hungarica*, 17(2), 2020, pp. 57-76

[20] Kővári A., Katona J., Costescu C.: Quantitative Analysis of Relationship Between Visual Attention and Eye-Hand Coordination, *Acta Polytechnica Hungarica*, 17(2), 2020, pp. 77-95

[21] Borus G. Gy., Gilányi A.: Computer assisted solution of systems of two variable linear functional equations, *Aequationes mathematicae*, 94(4), 2020, pp. 723-736

[22] Borus G. Gy., Gilányi A.: On a computer program for solving systems of functional equations, Proceedings of the 4th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Budapest, Hungary, 2013, p. 939

[23] Borus G. Gy., Gilányi A.: Solving systems of linear functional equations with computer, Proceedings of the 4th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Budapest, Hungary, 2013, pp. 559-562

[24] Gilányi A.: Solving linear functional equations with computer, *Math. Pannon.*, Vol. 9, No. 1, 1998, pp. 57-70

[25] Gilányi A., Merentes N., Quintero R.: Presentation of an animation of the m-convex hull of sets, Proceedings of the 7th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Wrocław, Poland, 2016, pp. 307-308

[26] Paulhus M.: An Algorithm for Packing Squares. *J. Combin. Theory Ser. A.*, 82, 1997, pp. 147-157

[27] Meir A., Moser, L.: On packing of squares and cubes. *J. Combin. Theory Ser. A.*, 5, 1968, pp. 126-134

[28] Brass P., Moser W. O. J., Pach J.: *Research Problems in Discrete Geometry*, Springer-Verlag, New York, 2005, pp. 121-122

[29] Jennings D.: On packing unequal rectangles in the unit square. *J. Combin. Theory Ser. A.*, 68, 1994, pp. 465-469

[30] Ball K.: On packing unequal squares. *J. Combin. Theory Ser. A.*, 75(2), 1996, pp. 353-357

[31] Joós A.: On packing of rectangles in a rectangle. *Discrete Math.*, 341(9), 2018, pp. 2544-2552

[32] Grzegorek P., Januszewski, J.: A note on three Moser's problems and two Paulhus' lemmas. *J. Comb. Theory, Ser. A.*, 162, 2019, pp. 222-230

Convexity and Mathability

Antal Joós [1], Attila Kovari [2,3,4]

[1]University of Dunaújváros, Institute of Computer Engineering, Department of Mathematics, Táncsics M. u. 1/a, 2400 Dunaújváros, Hungary, joosa@uniduna.hu

[2]University of Dunaújváros, Institute of Engineering, Department of Natural Science, Táncsics M. u. 1/a, 2400 Dunaújváros, Hungary, kovari@uniduna.hu

[3]Óbuda University, Alba Regia Technical Faculty, Budai út 45, 8000 Székesfehérvár, Hungary, kovari.attila@amk.uni-obuda.hu

[4]John von Neumann University, GAMF Faculty of Engineering and Computer Science, Izsáki u. 10, 6000 Kecskemét, Hungary, kovari.attila@gamf.uni-neumann.hu

Abstract: Mathability refers to a branch of cognitive infocommunications that investigates any combination of artificial and natural cognitive capabilities relevant to mathematics, including a wide spectrum of areas ranging from low-level arithmetic operations to high-level symbolic reasoning. In connection with investigations related to mathability and to applications of computer-assisted methods for studying mathematical problems, in this paper, animation of the planar hyperconvex sets of radius r is presented. This animation helps us understand some properties of hyperconvex sets and to see the differences between convexity and hyperconvexity.

Keywords: Mathability; Cognitive Infocommunications; Computer Assisted Methods; Animation; Convex set; Hyperconves set of raduis r; Spindle convex set; Ball polyhedron

1 Introduction

Mathability refers to a branch of cognitive infocommunications that investigates any combination of artificial and natural human cognitive capabilities relevant to mathematics, including a wide spectrum of areas ranging from low-level arithmetic operations to high-level symbolic reasoning. The concept was introduced in the paper [1] related to the 4th IEEE International Conference on Cognitive Infocommunications (CogInfoCom) in 2013. Mathability refers to devices with high mathematical and logical potential and is defined as human mathematical ability [8]. Mathability mainly discusses what new assimilation methods are used to process information and how people use this ability to build their knowledge using problem-solving and experiences as well as high-level mathematics applications [8]. Its educational aspects were investigated, among others, in [5]-[13], while [14]-[18] papers focus on human cognitive related aspects of CogInfoCom and how people can communicate with machines to possess new knowledge. Questions related to mathability and to computer-based methods for investigations of mathematical problems have been studied by several authors during recent years [19]-[23]. Computer-aided solutions of mathematical problems were presented in [22], [26] and some of its further general properties were described in the papers [2] and [3] and in the book [4]. In this paper, we also would like to contribute to these investigations. We present a computer-assisted method for a visualization related to the so-called hyperconvex discs of radius r.

Although convexity is one of the oldest concepts in geometry, it is used to investigate some modern phenomena in mathematics, i.e. this property is used in the qualitative theory of differential equations as well [24], [25].

The students meet the convexity several times during the education. In this paper, the basics of convexity and some generalizations of convexity will be introduced. The paper [26] motivated us to write an animation of hyperconvex discs of radius r. The animation is developed in GeoGebra available at https://www.geogebra.org/.

The hyperconvexity is a generalisation of convexity. Such kinds of generalisation of convexity shoves us the deeper attributes of convexity. The presented animation helps us understand some properties of hyperconvex sets and to see the differences between convexity and hyperconvexity.

2 The Convexity

The n-dimensional Euclidean space is denoted by $\mathbb{R}^n$. The notation $\cap$ means the intersection of sets. The points and vectors are identified in a natural way. In this paper, xy will also denote the length of the segment xy.

In this section we write the basic concepts of convexity.

In the school of geometry, a figure is called convex if it contains all segments if the endpoints of the segments lie in the figure. The next definition is the same.

Definition 2.1. The set C is *convex* if $x, y \in C$ implies that for any $\lambda \in [0,1]$ we have

$$\lambda x + (1-\lambda)y \in C. \tag{1}$$

Definition 2.2. The set N is *non-convex* if it is not convex, i.e. there are at least two points $x, y \in N$, and a λ number ($\lambda \in [0,1]$) such that

$$\lambda x + (1-\lambda)y \notin N. \tag{2}$$

Example 2.1. In Figure 1 we can see a convex set (left) and a non-convex set (right). Indeed, the right set is not convex, the midpoint m of the segment pq does not lie in the set.

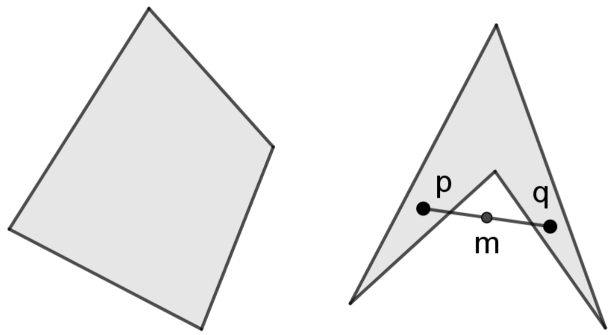


Figure 1
A convex set (left) and a non-convex set (right)

Example 2.2. The empty set is convex. The straight line segment is convex. The whole plane is convex.

The convex combination of the two points x and y is the set $\lambda x + (1-\lambda)y$ for all $\lambda \in [0,1]$ which is the straight line segment connecting the points x and y. The convex combination of finitely many points is the following.

Definition 2.3. The *convex combination* of the points $x_1, \dots, x_k \in \mathbb{R}^n$ is the linear combination

$$\lambda_1 x_1 + \cdots + \lambda_k x_k \tag{3}$$

for

$$\lambda_1 \geq 0, \dots, \lambda_k \geq 0 \text{ and } \lambda_1 + \cdots + \lambda_k = 1. \tag{4}$$

Example 2.3. The set of all convex combinations of two different points is a straight line segment. The set of all convex combinations of three non-collinear points is a triangle.

Remark 2.1. The set C is convex if and only if all the convex combinations of the points $x_1, \dots, x_k \in \mathbb{R}^n$ lie in the set C.

Theorem 2.1. The intersection of convex sets is convex.

This implies the next definition.

Definition 2.4. The *convex hull* of the set $S \in \mathbb{R}^n$ is the intersection of all convex sets containing S.

Remark 2.2. The convex hull of a set is convex.

Example 2.4. The convex hull of a convex set C is C.

Example 2.5. The convex hull of three different non-collinear points x_1, x_2, x_3 is the triangle of vertices x_1, x_2, x_3.

Definition 2.5. The *Minkowski sum* of the sets $S_1, S_2 \in \mathbb{R}^n$ is the set

$$\{s_1 + s_2 : s_1 \in S_1, s_2 \in S_2\}. \quad (5)$$

Notation: $S_1 + S_2$.

Example 2.6. The Minkowski sum of two non-parallel segments is a parallelogram.

Theorem 2.2. Let C_1 and C_2 be two convex sets in R^n. The translation $C_1 + p$ is convex. The scaling αC_1 is convex. The orthogonal projection of the set C_1 is convex. The Minkowski sum $C_1 + C_2$ is convex.

3 Some Generalizations of Convexity

3.1 The m-convexity

Toader [27] introduced the m-convexity in the following way.

Definition 3.1.1. Let $m \in [0,1]$ be a fixed number. The set $C \in R^n$ is m-convex if

$$tx + m(1-t)y \in C \quad (6)$$

for all elements $x, y \in C$ and for each $t \in [0,1]$.

Example 3.1.1. If $m = 1$, then the definitions convex and m-convex are the same.

It is a consequence of the definition, that if $m \neq 1$, then it is necessary to consider the origin as well.

Example 3.1.2. If $m = 0.5$, then the m-convex set containing the points x and y contains the point $\frac{m}{m+1}(x + y)$ as well (see, e.g. [26])

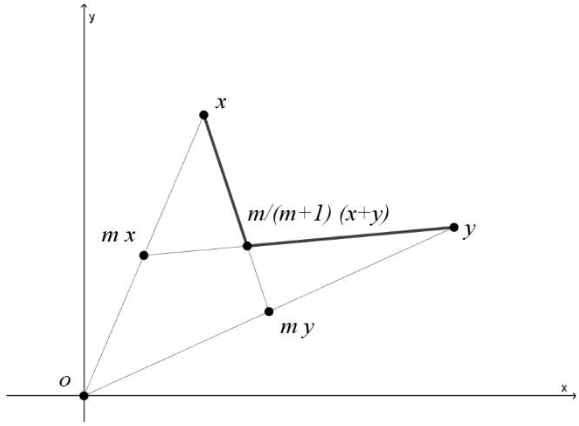


Figure 2
The line connecting the points x and y in an m-convex set

Definition 3.1.2. The *m-convex combination* of the points $x_1, \dots, x_k \in \mathbb{R}^n$ is the linear combination

$$\lambda_1 x_1 + m(\lambda_2 x_2 + \cdots + \lambda_k x_k) \tag{7}$$

for

$$\lambda_1 \geq 0, \dots, \lambda_k \geq 0 \text{ and } 0 < \lambda_1 + \cdots + \lambda_k \leq 1. \tag{8}$$

Definition 3.1.3. The *m-convex hull* of the set $S \in \mathbb{R}^n$ is the intersection of all m-convex sets containing S.

Theorem 3.1.1 [28] The set S is m-convex if and only if S is the set of all m-convex combinations of points lying in S.

Theorem 3.1.2 [28] The m-convex hull of the set S is the set of all m-convex combinations of points lying in S.

Theorem 3.1.3 [28] Let S be a set containing the origin o. The set S is m-convex if and only if for all $x, y \in S$ the set $conv\left(o, x, \frac{m}{m+1}(x+y)\right) - \{o\}$ is contained in S.

Example 3.1.3. The m-convex hull of the two different points x, y and the origin o is the (degenerate) quadrangle $ox\frac{m}{m+1}(x+y)y$ (Figure 2).

In [21] and [25] we can find an animation of m-convex hull of finitely many points if m is varied. This paper motivated us to produce a similar animation for hyperconvex sets of radius r.

3.2 The Hyperconvex Sets of Radius r

Definition 3.2.1. The *n-dimensional ball* (or shortly *n-ball*) of radius r and center c in R^n, denoted by $B(r, c)$, is $\{x \in R^n: xc \leq r\}$. If $n = 2$, then the ball is called *disc*.

Definition 3.2.2. The *n-dimensional sphere* (or shortly *n-sphere*) of radius r and center c in R^n is $\{x \in R^n: xc = r\}$. If $n = 2$, then the sphere is called *circle*.

Definition 3.2.3. Let $x, y \in R^n$. If $xy < 2r$, then the *spindle of radius* r (or shortly *spindle) of* x *and* y is defined as the union of circular arcs with endpoints x and y that are of radii at least r and shorter than a semicircle of radius r. If $xy = 2r$, then the *spindle* of x *and* y is defined as the disc of radius r and center $(x + y)/2$. If $xy > 2r$, then the *spindle* of x *and* y is defined as $\emptyset$.

Remark 3.2.1. The spindle of x and y is the intersection of the balls of radii r and containing x and y.

Example 3.2.1. In Figure 3 can be found a spindle of radius 1 of x and y on the plane if $xy < 2$.

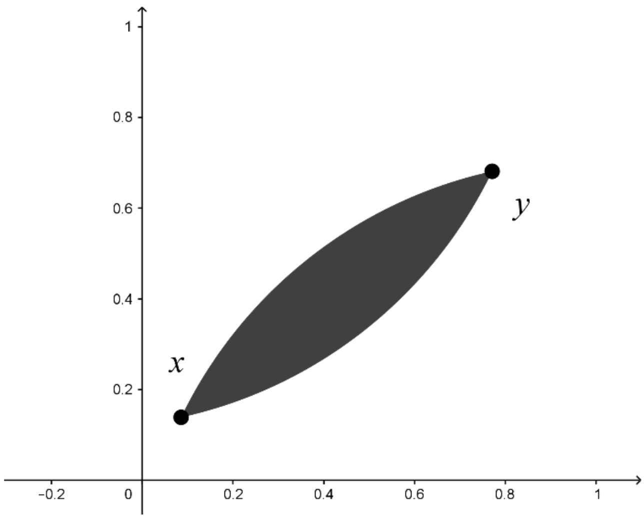


Figure 3
The spindle of x and y

Definition 3.2.4. Let C be the set such that the diameter of C is less than or equal to $2r$. The set C is *spindle convex* (of radius r) if $x, y \in C$ implies that the spindle of x and y is a subset of C.

Definition 3.2.5. The *circumradius*, denoted by $cr(C)$ of a bounded set C in R^n is defined as the radius of the unique smallest ball that contains C. If C is unbounded, then $cr(C) = \infty$.

Definition 3.2.6. A set C in R^n is *hyperconvex of radius* r (or shortly *hyperconvex*) if is the intersection of n-balls of radius r.

Remark 3.2.2. Observe if we consider half-spaces as balls of infinite radius, then the hyperconvexity of radius r and (linear) convexity are the same.

Definition 3.2.7. Let C be a finite set in R^n such that $cr(C) \leq r$. The *ball-polyhedron of radius* r (or shortly *ball-polyhedron*) (generated by C) is the intersection of the balls for radii r and centers of points in C. If $n = 2$, then a ball-polyhedron is called a *disk-polygon*.

Observe the ball-polyhedron of radius r generated by C is

$$P = \bigcap_{c \in C} B(r, c). \quad (9)$$

First Mayer [29] considered ball-polyhedra in 1935 and called this property "überkonvex". Mayer's paper inspired several researchers in the first half of the 20th Century e.g. [30]-[36]. 1980's we can find this property as r-convex or spindle convex of hyperconvex of radius r see, e.g. [37]-[47].

Definition 3.2.8. If a ball B contains a set C in R^n and a point x lies on the boundary of B and the boundary of C at the same time, then B supports C at x.

Theorem 3.2.1. Let C be a closed convex set in R^n such that $cr(C) \leq r$. The following are equivalent.

1) The set C is spindle convex of radius r.
2) The set C is the intersection of unit balls of radius r containing C.
3) For every boundary point of C, there is a ball of radius r that supports C at that point.

Definition 3.2.9. The *hyperconvex hull of radius* r of the set $S \in \mathbb{R}^n$ is the intersection of all hyperconvex sets of radius r containing S.

4 The Description of Animation

We use the dynamic free software GeoGebra, which can be downloaded from https://www.geogebra.org/.

To have a GeoGebra file, which can be easily modified, we use scripts under buttons.

The first GeoGebra script under button1 in On Click is the input of points, a list which consists of the points, the default value of r, and a text. In this special case, we use six points.

```
P1=(20,5); P2=(13,6.5); P3=(17,13); P4=(12,16); P5=(7,17); P6=(0.5,7)
L={P1,P2,P3,P4,P5,P6}
r=1
text1= "The diameter is larger than 2r."
```

The JavaScript under button2 in On Click is the drawing of the hyperconvex set of radius r generated by the points in the list L. The code is the following.

```
1 D=ggbApplet.getValue("length(L)");
2 ggbApplet.evalCommand("LSegm_{0}={}"); k=1;
3 for(var i =1;i<D+1;i++) for(var j=i+1;j<D+1;j++) {
4 ggbApplet.evalCommand("LSegm_{"+k+"}=Append(LSegm_{"+(k-
1)+"},Segment(L("+i+"),L("+j+")))");
5 k=k+1; }
6 ggbApplet.evalCommand("Diam=Max(LSegm_{"+(k-1)+"})");
7 ggbApplet.evalCommand("Conv=ConvexHull(L)");
8 ggbApplet.evalCommand("ShowLabel(Conv,False)");
9 for(var i =1;i<D+1;i++) {
10 ggbApplet.evalCommand("C_{"+i+"}=Circle[L("+i+"),r]");
11 ggbApplet.evalCommand("SetVisibleInView(C_{"+i+"},1,False)"); }
12 for(var i =1;i<D+1;i++) for(var j =i+1;j<D+1;j++) {
13 ggbApplet.evalCommand("Center_{"+i+","+j+",1}=Intersect[C_{"+i+"}, C_{"+j+"},1]");
14 ggbApplet.evalCommand("SetVisibleInView(Center_{"+i+","+j+",1},1, False)");
15 ggbApplet.evalCommand("Center_{"+i+","+j+",2}=Intersect[C_{"+i+"}, C_{"+j+"},2]");
16 ggbApplet.evalCommand("SetVisibleInView(Center_{"+i+","+j+",2},1, False)");
17 ggbApplet.evalCommand("ineq_{"+i+","+j+",1}=(x-x(Center_{"+i+","+j+",1}))^2+(y-
y(Center_{"+i+","+j+",1}))^2-r^2<=0");
18 ggbApplet.evalCommand("SetVisibleInView(ineq_{"+i+","+j+",1},1, False)");
19 ggbApplet.evalCommand("ineq_{"+i+","+j+",2}=(x-x(Center_{"+i+","+j+",2}))^2+(y-
y(Center_{"+i+","+j+",2}))^2-r^2<=0");
20 ggbApplet.evalCommand("SetVisibleInView(ineq_{"+i+","+j+",2},1, False)");
21 ggbApplet.evalCommand("ineq_{"+i+","+j+"}=ineq_{"+i+","+j+",1}&&
ineq_{"+i+","+j+",2}");
22 ggbApplet.evalCommand("SetLineThickness(ineq_{"+i+","+j+"},0)");
23 ggbApplet.evalCommand('SetColor(ineq_{'+i+','+j+'},"#000000")');
24 ggbApplet.evalCommand("ShowLabel(ineq_{"+i+","+j+"},False)");
25 ggbApplet.evalCommand("Delete(eq1)");
26 ggbApplet.evalCommand("Delete(eq2)"); };
27 ggbApplet.evalCommand("If(Diam>2*r,SetVisibleInView(text1,1,
True),SetVisibleInView(text1,1,False))");
```

This is a rough algorithm. Our aim was visualization and not effectiveness. The first line adds the value of the length of the list L to the JavaScript code. Lines 2-6 determine the diameter of the point set lying in the list L. Since GeoGebra is a dynamic language the modification of the points lying in L implies the modification of the diameter calculated by this code in real-time. Lines 7-8 produce the convex hull of the point set lying in the list L. Lines 9-26 define the spindles of the pairs of points in the list L. Line 27 hides the text text1 if the diameter of L is less than or equal to 2r. The result is the hyperconvex hull of radius r generated by the points in L. If we let show the object r as a slide between 0.01 and 5, then we can vary the radius r in the hyperconvex set. If we switch on the animation of r, then in Figures 4 and 5 we can see a selection of six stages of the hyperconvex hull of radius r of the list of points given for the values $r = 1, 1.2, 1.4, 1.6, 1.8$ and 5.

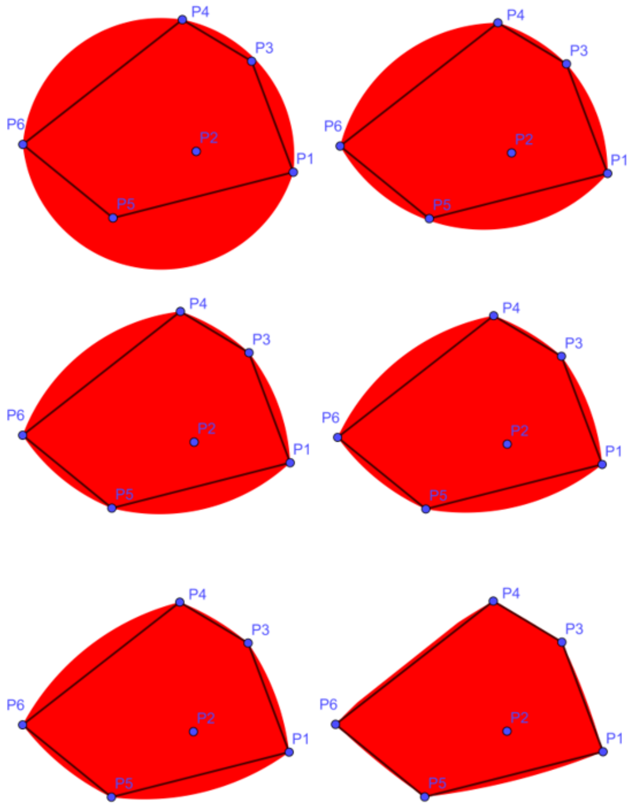


Figure 4

Some stages from the animation

Conclusion

The results presented here are connected to the investigations of mathability (cf. [1] and [4].) Nowadays hyperconvexity is a popular generalisation of convexity in the literature of discrete and convex geometry (see e.g. [43]).

Convexity is a wide range applicable concept in mathematics. The first step considering hyperconvex sets is drawing such a set on a piece of paper. This dynamic animation enables us to draw this figure. In order to imagine a hyperconvex set in higher dimensions we have to understand the planar case.

Since a generalization of convexity can be challenging to imagine or understand, it is important to visualize hyperconvexity. We can find serious theorems in the literature considering the difference between linear convexity and a generalization of convexity. If we use such simple animations, then we can make conjecture about new theorems and about the difference between the two kinds of convexity easier.

The presented method inspires us to visualize geometric properties of point sets in GeoGebra. GeoGebra supports the script commands as we can see in this paper. It could be a different opportunity to create a new tool in GepGebra to visualize the hyperconvex hull of a finite point set.

Acknowledgement

This work was supported by EFOP-3.6.1-16-2016-00003 funds. Establishment of long-term R and D and I processes at the University of Dunaújváros.

References

[1] Baranyi P., Gilányi A.: Mathability: emulating and enhancing human mathematical capabilities, Proceedings of the 4th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Budapest, Hungary, 2013, pp. 555-558

[2] Baranyi P., Csapo A., Varlaki P.: An overview of research trends in coginfocom. Proceedings of the 18th International Conference on Intelligent Engineering Systems (INES), Tihany, Hungary, 2014, pp. 181-186

[3] Török M., Tóth J., Szöllősi A.: Foundations and perspectives of mathability in relation to the coginfocom domain, Proceedings of the 4th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Budapest, Hungary, 2013, pp. 869-872

[4] Baranyi P., Csapo A., Sallai G.: Cognitive Infocommunications (CogInfoCom) Springer, 2015

[5] Chmielewska K. et al: Learnability – are we ready for distance learning?, Proceedings of the 11th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Online, 2020, pp. 465-470

[6] Biró P., Csernoch M.: Deep and surface metacognitive processes in non-traditional programming tasks, Proceedings of the 5th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Vietri sul Mare, Italy, 2014, pp. 49-54

[7] Kővári A., Rajcsányi-Molnár M.: Mathability and Creative Problem Solving in the MaTech Math Competition. Acta Polytechnica Hungarica, 17(2), 2020, pp. 147-161

[8] Chmielewska K., Gilanyi A.: Educational context of mathability, Acta Polytechnica Hungarica, Vol. 15, No. 5, 2018, pp. 223-237

[9] Biró P., Csernoch M.: Deep and surface metacognitive processes in non-traditional programming tasks, Proceedings of the 5th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Vietri sul Mare, Italy, 2014, pp. 49-54

[10] Chmielwska K. et al: Mathability and mathematical cognition, Proceedings of the 7th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Wroclaw, Poland, 2016, pp. 245-250

[11] Szi B., Csapo A.: An outline of some human factors contributing to mathability research, Proceedings of the 5th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Vietri sul Mare, Italy, 2014, pp. 583-586

[12] Chmielwska K., Gilányi A.: Mathability and computer aided mathematical education, Proceedings of the 6th IEEE Conference on Cognitive Infocommunications (CogInfoCom) Győr, Hungary, 2015, pp. 473-477

[13] Piró P., Csernoch M.: The mathability of computer problem solving approaches. Proceedings of the 6th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Győr, Hungary, 2015, pp. 111-114

[14] Katona J. et al: Using Eye- Tracking to Examine Query Syntax and Method Syntax Comprehension in LINQ, Proceedings of the 12th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Naples, Italy, 2020, pp. 437-444

[15] Kővári, A., Katona, J., Costescu, C.: Evaluation of Eye-Movement Metrics in a Software Debugging Task using GP3 Eye Tracker. Acta Polytechnica Hungarica, 17(2), 2020, pp. 57-76

[16] Námesztovszky, Z., & Boros, O. (2019). The Implementation of Projects which Develop Both Soft and Interdisciplinary Skills Using Micro:bit. Journal of Applied Technical and Educational Sciences, 9(2), 42-51

[17] Kővári A., Katona J., Costescu C.: Quantitative Analysis of Relationship Between Visual Attention and Eye-Hand Coordination. Acta Polytechnica Hungarica, 17(2), 2020, pp. 77-95

[18] Orosz, B. et al: Digital education in digital cooperative environments. Journal of Applied Technical and Educational Sciences, 9(4), 2019, 55-69

[19] Borus G. Gy. et al: On a computer program for solving systems of functional equations, Proceedings of the 4th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Budapest, Hungary, 2013, p. 939

[20] Gilányi A.: Solving linear functional equations with computer, Math. Pannon., Vol. 9, No. 1, 1998, pp. 57-70

[21] Borus G. G., Gilányi A.: Computer assisted solution of systems of two variable linear functional equations. Aequationes mathematicae, 94(4), 2020, pp. 723-736

[22] Gilányi A., Merentes N., Quintero R.: Presentation of an animation of the m-convex hull of sets, Proceedings of the 7th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Wrocław, Poland, 2016, pp. 307-308

[23] Borus G. Gy. et al: Solving systems of linear functional equations with computer, Proceedings of the 4th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Budapest, Hungary, 2013, pp. 559-562

[24] Nagy B.: Analysis of the Biological Clock of Neurospora. J. Comp. Appl. Math., 2009, pp. 298-305

[25] Simon P., Farkas H., Wittmann M.: Constructing global bifurcation diagrams by the parametric representation method. J. Comp. Appl. Math., 1999, pp. 157-176

[26] Gilányi A., Merentes N., Quintero R.: Mathability and an animation related to a convex-like property, Proceedings of the 7th IEEE Conference on Cognitive Infocommunications (CogInfoCom), Wroclaw, Poland, 2016, pp. 227-232

[27] Toader G.: Some generalizations of the convexity. Proceedings of the Colloquium on Approximation and Optimization, Univ. Cluj-Napoca, Cluj-Napoca, 1985, pp. 329-338

[28] Lara T.: On m-Convexity on Real Linear Spaces. UPI Journal of Mathematics and Biostatistics, 2018, pp. 1-16

[29] Mayer, A. E.: Hyperconvex sets (Eine Überkonvexitat). Math. Z., 39, 1935, pp. 511-531

[30] Blanc E.: Sets on convex planes (Les ensembles surconvexes plans). Ann. Sci. Ecole Norm. Sup., 60 (3), 1943, pp. 215-246

[31] Buter J.: About convex sets in the plane (Überconvexe Mengen in der Ebene) Ak. Wetensch. Amsterdam Proc. 41, 1938

[32] Pasqualini L.: Superconvexity (Superconvexité). Bull. Cl. Sci. Acad. Belg., 25 (5), 1939, pp. 18-24

[33] Santaló L. A.: On plane hyperconvex figures. Summa Brasil. Math., 1, 1946, pp. 221-239

[34] Corput J. G.: About convex sets in the plane (Überkonvexe Mengen in der Ebene). Ak. Wetensch. Amsterdam Proc., 41, 1938

[35] Vincensini P.: Superconvex figures in the plane (Sur les figures superconvexes planes). Bull. Soc. Math. France, 64, 1936, pp. 197-208

[36] Danzer L., Grünbaum B., Klee V.: Helly's theorem and its relatives. Proc. Sympos. Pure Math., Vol. VII, Amer. Math. Soc., Providence, R. I., 1963, pp. 101-180

[37] Fejes Tóth L.: Packing of r-convex discs. Studia Sci. Math. Hungar, 17, 1982, pp. 449-452

[38] Fejes Tóth, L.: Packing and covering with r-convex discs. Studia Sci. Math. Hungar, 18, 1982, pp. 69-73

[39] Bezdek K., Lángi. Zs, Naszódi M., Papez P.: Ball-polyhedra. Discrete Comput. Geom., 2007, pp. 201-230

[40] Kupitz Y, Martini H., Perles M.: Ball polytopes and the Vázsonyi problem. Acta Math. Hungar, 126, 2010, pp. 99-163

[41] Kupitz Y, Martini H., Perles M.: Finite sets in R^d with many diameters — a survey. Proceedings of the International Conference on Mathematics and Applications (ICMA-MU 2005, Bangkok), Mahidol University Press, Bangkok (Reprinted in a special volume of the East-West J. Math.: Contributions in Mathematics and Applications), 2005, 2007, pp. 91-112 (41-57)

[42] Bezdek K.: Lectures on sphere arrangements-the discrete geometric side, Fields Institute Monographs. New York: Springer, 2013

[43] Bezdek K.: Classical topics in discrete geometry, CMS Books in Mathematics/Ouvrages de Math´ematiques de la SMC. New York: Springer, 2010

[44] Fodor F., Fejes Tóth G.: Dowker-type theorems for hyperconvex discs. Period. Math. Hungar, 70, 2015, pp. 131-144

[45] Fodor F., Vígh V.: Variance estimates for random disc-polygons in smooth convex discs. arXiv, 2018

[46] Fodor F.: Inequalities for hyperconvex sets. Adv. Geom., 2016, pp. 337-348

[47] Bezdek A., Joós A.: Area minimization of special polygons. Acta Math. Hungar., 160 (1), 2020, pp. 33-44

The Mathability of Computer Problem Solving with ProgCont

Piroska Biró[1], Tamás Kádek[2]

[1]University of Debrecen, Faculty of Informatics, Kassai út 26, 4028 Debrecen, Hungary, biro.piroska@inf.unideb.hu

[1]Sapientia Hungarian University of Transylvania, Faculty of Economics, Socio-Human Sciences and Engineering, Miercurea-Ciuc, Romania, biropiroska@uni.sapientia.ro

[2]University of Debrecen, Faculty of Informatics, Kassai út 26, 4028 Debrecen, Hungary, kadek.tamas@inf.unideb.hu

Abstract: In the teaching/training of programmers, the development of mathematical skills is given as much priority as that of IT skills. A complex IT problem solution is inconceivable without adequate mathematical background knowledge. In our research, we would like to show this connection through the analysis of programming tasks, for which we use the automatic solution evaluation system developed by the Faculty of Informatics of the University of Debrecen almost a decade ago and the data accumulated during its use. We examine the effectiveness of users in solving programming tasks that require mathematical and IT knowledge, compare performance in different programming languages, and look for topics where improvement is needed. The results show that, contrary to our expectations, students perform better on tasks that require both mathematics and informatics skills.

Keywords: ProgCont system; automated evaluation; programming; mathability; computer problem solving

1 Introduction

At the Faculty of Informatics of the University of Debrecen, teaching mathematics and informatics subjects goes hand in hand with the various IT majors (Computer Science, Computer Science Engineering, Business Informatics). Knowledge in the relevant fields of both disciplines is essential for the successful completion of these majors. The two disciplines are also often intertwined within subjects. Subjects aimed at acquiring programming skills, from subjects introducing algorithmic thinking to subjects describing high-level programming languages, are no exception in this respect [12], [20]. In teaching different programming languages, program writing tasks are typically used to measure students' knowledge. In the case of these tasks, both mathematical and IT knowledge are required, albeit to a different extent, depending on the type of the task. In our research, we would like to understand the relationships and differences between these types of tasks by examining the effectiveness of our students in solving tasks [11], [19].

One of the basic goals of the ProgCont system developed at the University of Debrecen, Faculty of Informatics since 2011, is to develop students' mathematical and IT skills and to promote students' learning and practice. The system primarily assists in the teaching of programming languages by testing and evaluating programs submitted in different programming languages [13]-[17].

Based on nearly a decade of experience and use, we can say that such a system and tool has been developed that also improves students' mathability skills [4]-[10], [18]. The concept of mathability [2] was born under Cognitive Infocommunications [1], [3].

In this article, we compare student performance measured on tasks requiring mathematics and informatics skills over seven years. In our research, we use three hypotheses to look for correlations between these abilities.

2 Research

2.1 The ProgCont System

The ProgCont system has been developed and used by the Faculty of Informatics from 2011 to the present day for automatic evaluation of programming tasks [13]-[17]. The software is very similar to the Moodle CodeRunner plugin. It compiles, runs, and checks solutions for programming tasks submitted in various source languages (C, C++, C#, Java, Pascal, Python), comparing their output with the expected correct output.

The system was developed primarily for programming competitions to objectively evaluate competition tasks. However, it soon became clear that it could also be used in examinations and preparing for examinations. Over the last decade, the system has been growing steadily. More and more different tasks have been formulated, and it has become possible to support more and more programming languages. Up to now 45 competition problem sets, 241 examination problem sets, 11 practice problem sets are available in the system with a total of 1 657 tasks.

At first glance, the number of practice problem sets may seem remarkably low compared to the examination tasks, however, the examination tasks remain available for practice after the examination. In addition to the usual programming languages (C, C++, Java, Pascal) in international ACM-style programming competitions, support for the C# programming language soon appeared, and the list has been recently expanded to include Python.

Table 1
Table of all submissions by programming languages until 31/08/2020

Programming language	Submissions	Accepted	Pass rate
ANSI C	68 201	19 257	28 %
C (C99)	118 992	40 925	34 %
C++	9 983	2 752	28 %
C#	19 892	8 703	44 %
Java	58 258	22 237	38 %
Pascal	1 366	150	11 %
Python	418	99	24 %
Total	**277 110**	**94 123**	**34 %**

Due to the diverse nature of the use, the range of users is also very wide:

– high school students preparing for, or participating in, competitions;
– participants of extracurricular activities for high school students organized by our faculty;
– IT students of our faculty preparing for, or participating in, competitions;
– students preparing for and taking examinations in, certain subjects.

The ProgCont system has been developed primarily based on non-pedagogical aspects, but at the same time it is playing an increasingly important role in supporting education, and in the last 10 years so much data has been collected that should be analysed with pedagogical methods.

2.2 Hypotheses

The present research aims to use the database, which has been growing for almost a decade, to examine and compare the IT and mathematical skills of the widest possible range of users and the possible connection between them. The following hypotheses were defined:

Hypothesis 1: *Users of our system perform better in solving tasks that require only IT skills than those where both IT and mathematical competencies are required.*

Hypothesis 2: *A set of publicly available tasks in the ProgCont system can be identified where users perform poorly (we can suggest types of problems where more tasks should be set up to improve user performance).*

Hypothesis 3: *There exists a relationship between the type of task and the programming language chosen for its solution that can be observed among our users.*

2.2.1 SWOT Analysis

Strengths

A database is available that contains a significant number of tasks (1 657) and objective (automatically performed) evaluation of a large number (277 108) of the submitted solutions.

Weaknesses

Task solutions cannot be bound to individuals. We have deliberately kept the ProgCont system, especially since the release of GDPR, away from storing information that requires the processing of personal data. Thus, for example, the student whom a solution submitted for evaluation belongs to during an assessment is managed by an external system, and thus, no information is available during the research. We can only make statements that are generally valid for ProgCont users.

Opportunities

The pandemic has given a huge boost to the development of online education and has, in the eyes of many, appreciated the potential that resides in our ProgCont software. We can certainly hope that the number of users will increase further shortly, which will provide an opportunity to expand the present research in the future. Therefore, in the analyses examining the hypotheses discussed in this article, we have observed that this can be easily repeated in the future.

Threats

By adopting the provisions of GDPR, the change in the legal environment has not stopped. We often find that some institutions, including our university, transpose regulation into their practice in a stricter way. This requires constant adaptation.

2.2.2 The Research Sample

To test the hypotheses on as large a sample as possible, we chose tasks that have been available for some time and may have been of interest to a wide range of users. Therefore, we chose a collection of nearly 100 tasks that had already been collected in 2014 in the ProgCont system as the basis for our research Table 2.

The advantage of the collection is that the tasks are grouped based on 9 topics related to informatics or mathematics. As in ProgCont most of the problem sets contain previously published problems, we also examined evaluated solutions coming from the time before the collection was compiled. Due to all this, we chose seven years for the analysis, from 01/12/2012 to 01/12/2019. For all examined tasks, at least one solution was received before 01/12/2012, so all tasks were available to users during the examined period.

Since the aim is to compare the performance of solving tasks requiring IT and mathematical skills, we kept only those 3-3 of the 9 topics for which these skills can be well distinguished. We selected 3 informatics and 3 mathematics topics area (Table 2).

Category of mathematics tasks:

- Number theory (7 tasks)
- Geometry (5 tasks)
- Arithmetic and algebra (12 tasks)

Category of informatics tasks:

- Simulation (10 tasks)
- String operations (12 tasks)
- Sorting and searching (6 tasks)

Table 2
The sample and the categories

Category	Submissions	Accepted	Pass rate
Simulation	585	91	16 %
String operations	1 100	191	17 %
Sorting and searching	703	167	24 %
Informatics	**2 388**	**449**	**19 %**
Number theory	1 553	334	22 %
Geometry	371	83	22 %
Arithmetic and algebra	1 505	625	42 %
Mathematics	**3 429**	**1 042**	**30 %**
Total	**5 817**	**1 491**	**26 %**

In order not to distort the results, for example, the examination of tasks requiring knowledge of graph algorithms was omitted from the processing, as these included tasks that could be classified in both the former and the latter category or were difficult to classify. Tasks with a remarkably low number of submitted solutions (less than 20) were not processed either.

In the indicated period, a total of 5 817 submissions were received for the selected tasks, and 1 491 of them were correct, which means 26% of the total.

2.2.3 The Programming Task Categories

Based on Figure 1, it can be seen that the first hypothesis is not satisfied. There is a significant difference between the two groups, so the pass rate is higher for mathematics tasks than for informatics tasks (Figure 1 and Figure 2).

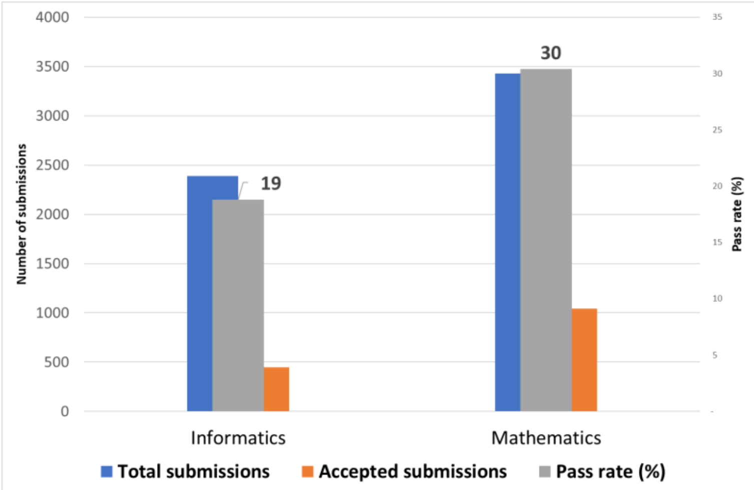


Figure 1
The acceptance rate in the two disciplines

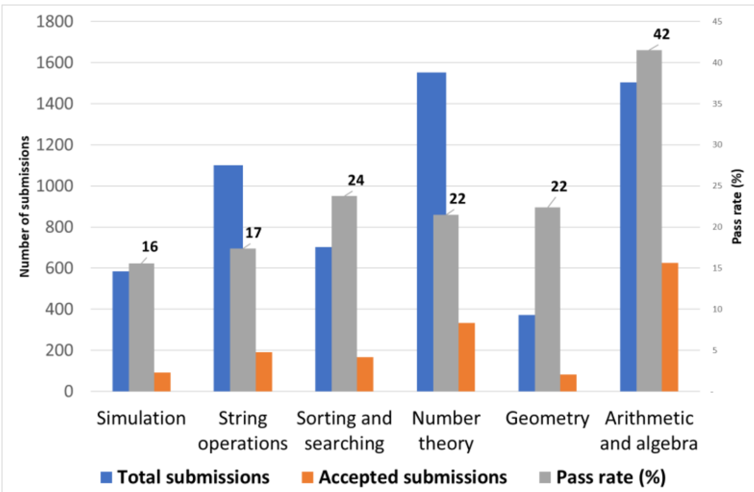


Figure 2
The acceptance rate in the selected categories

Students may be more confident in their knowledge of mathematics during programming learning than in their newly acquired IT skills; that is why submissions for tasks classified in the mathematics category are more successful. Greater emphasis needs to be placed on improving IT skills. To examine this in more detail, it is also worth looking at the distribution of submissions by tasks.

2.2.4 Error Distribution of Submitted Source Codes

The ProgCont system can provide six possible outputs when evaluating submissions, which can be used to categorize submissions with errors:

- *Compile error:* the submitted solution contains a syntax error, so the source code cannot be compiled.
- *Wrong answer:* the submitted solution produced incorrect output.
- *Presentation error:* the submitted solution produced incorrect output, but this differs from the expected output only in whitespace characters.
- *Runtime error:* An error occurred while running the submitted solution.
- *Time out:* the submitted solution did not run within the specified time limit.

In the research, we did not address solutions rejected with the compile error message, as these cannot be evaluated. The presentation error feedback has been in use since May 2017, before which such errors also resulted in wrong-answer feedback. Therefore, in our present research, we have merged the two categories and included them under the wrong answer tag.

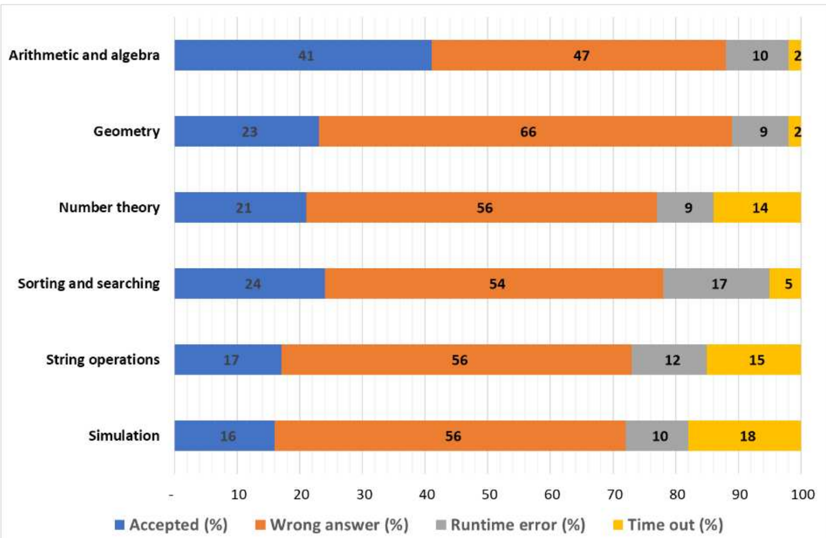


Figure 3
Distribution of programming errors in each programming task category

Figure 3 clearly shows that the most common type of error in submissions is the wrong answer in each category. It can also be seen from the figure that our users perform significantly worse in the case of tasks classified in the “String operations” and “Simulation” categories. Therefore, (confirming Hypothesis 2), it is necessary to expand the collection of tasks for these topics.

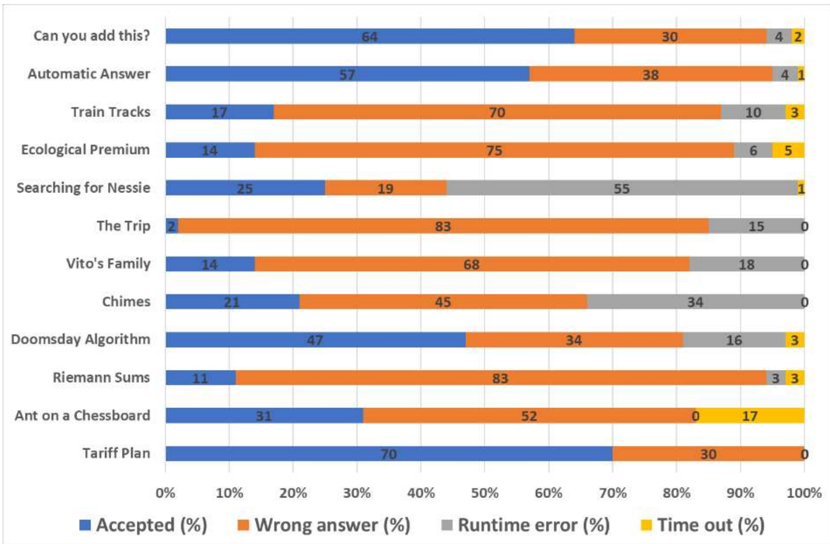


Figure 4
Distribution of programming errors by tasks in arithmetic and algebra

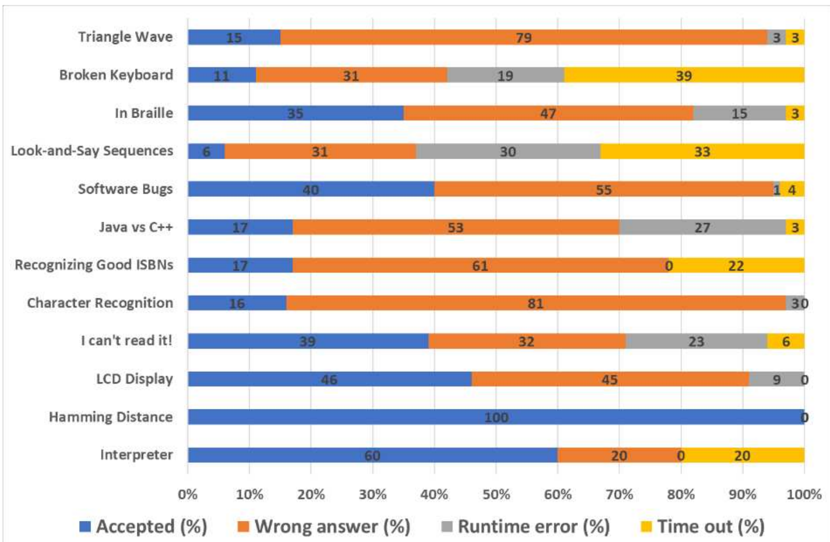


Figure 5
Distribution of programming errors by tasks in string operations

The difference between the typical errors in solving mathematical and IT problems is well shown by the comparison of task types "Arithmetic and algebra" and "String operations" with 12-12 tasks. The tasks were given in descending order of the number of solutions submitted. The four tasks “Tariff Plan”, “LCD Display”, “Hamming Distance”, and “Interpreter” are among the tasks that were omitted from the previous processing (less than 20 submissions). These are shown here (Figure 4 and Figure 5) only for the sake of completeness.

In addition to the overthrow of the first hypothesis, the number of occurrences of different error messages also shows surprising results. We thought that the “Time out” type of error would be more typical for mathematical problems. This is because a “Time out” error can typically be caused by someone trying to solve the problem as a simulation rather than using the appropriate solution formula. In light of this, it is quite surprising that this error is much more common for string operations than for arithmetic problems. It is difficult to distinguish between a wrong answer and a runtime error, for example, mistyping a single index variable in an array reference can result in either, depending on the specific case. Although we can conclude that the programs that correctly implement an incorrect algorithm fall into the wrong answer category. As can be seen from Figure 4 and Figure 5 this is more typical of mathematical problems.

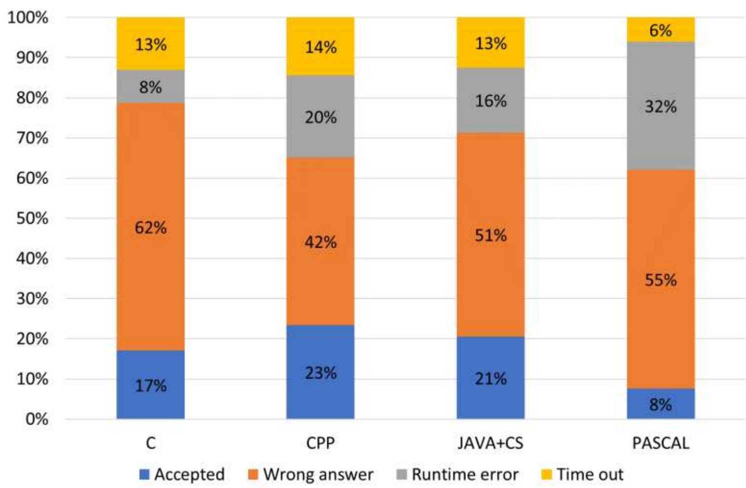


Figure 6

Distribution of programming errors by programming languages in tasks related to Informatics

Figure 6 and Figure 7 show the distribution of programming errors by the programming language in the two categories (informatics and mathematics). There were only a few submissions in the Python language (23 out of 5817) therefore we do not consider them. The comparison shows that students are more successful in solving informatics problems in an object-oriented language, while students are more successful in solving mathematics problems in C (Figure 6 and Figure 7).

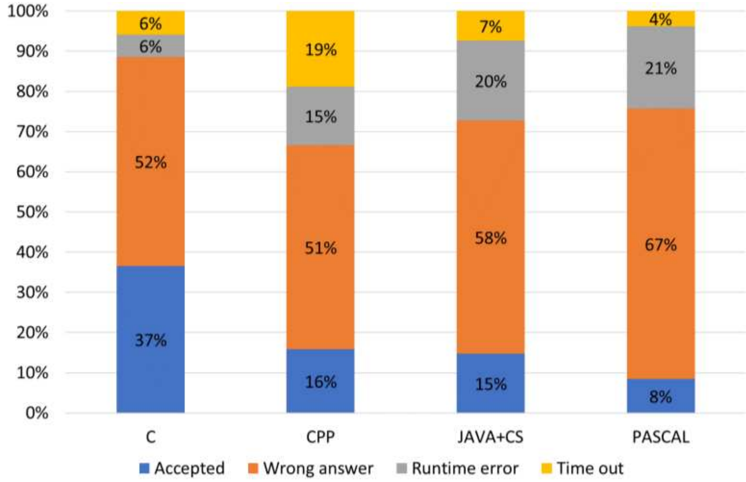


Figure 7
Distribution of programming errors by programming languages in tasks related to Mathematics

2.2.5 Distribution by Programming Languages

The ProgCont system can be used to submit source code for solving tasks in 7 different programming languages. These were divided into 5 categories:

- C: ANSI C (C89) and C (C99)
- CPP: C++,
- JAVA + CS: C# and Java,
- Pascal,
- Python.

Figure 5 shows the percentage distribution of submissions by language, and we have grouped submissions in Java and C# into one category (since IT also typically refers to these languages as one). Because there have been very few solutions in Python, we will not consider them in subsequent analyses. One reason for this is that Python language support did not exist until October 2016, while at least one programming language in the other 4 categories was available throughout the period under consideration.

74% of the submitted solutions were in C language, which correlates with the distribution of the programming languages taught in the University, because the subjects *Introduction to Programming* and *Programming Languages 1* focus on C language, and so the students practice more in this language.

This is especially important to keep in mind when examining the distribution of the preferred programming languages for math problems. Typically, the C language is not what we would recommend for solving mathematical tasks. Yet, it is strongly over-represented simply because our students only know this when they encounter these tasks.

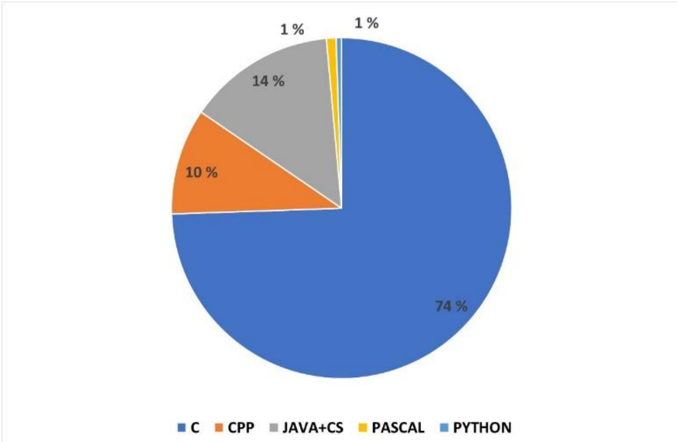


Figure 8
Distribution of submissions in different programming language

Figure 9 shows the distribution of programming languages in each of the two categories. The data show that our Hypothesis 3, according to which there is a programming language specific to the disciplinary field has been confirmed: C and Pascal are more common for mathematical tasks, and an object-oriented language (C++, Java, or C#) is more common for IT tasks (Figure 9 and Figure 10). This supports a recent decision to make students learn Python as their first language.

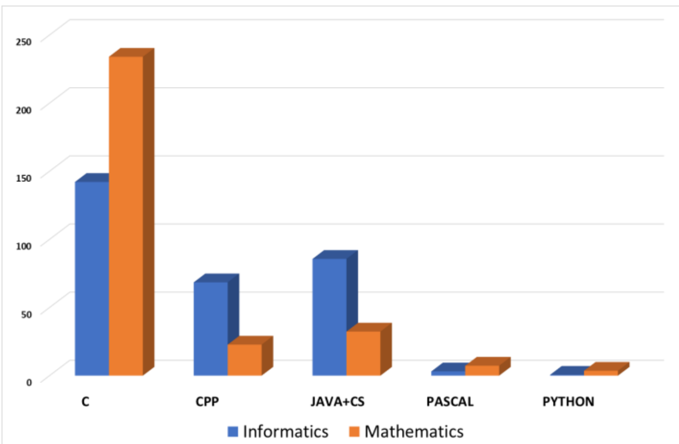


Figure 9
Distribution by programming languages in the two disciplines

The subject of our present study is only those tasks that could be solved in any programming language. Among our students, those who have already mastered an object-oriented language should also be able to program in at least one structured programming language. In their case, the choice of programming language may be more conscious, however, many students have dealt with these tasks who are not yet familiar with an object-oriented language, so it is not worth concluding the exceptionally high number of submissions in C. (The ratio of submission numbers is only interesting in comparing the two types of tasks.)

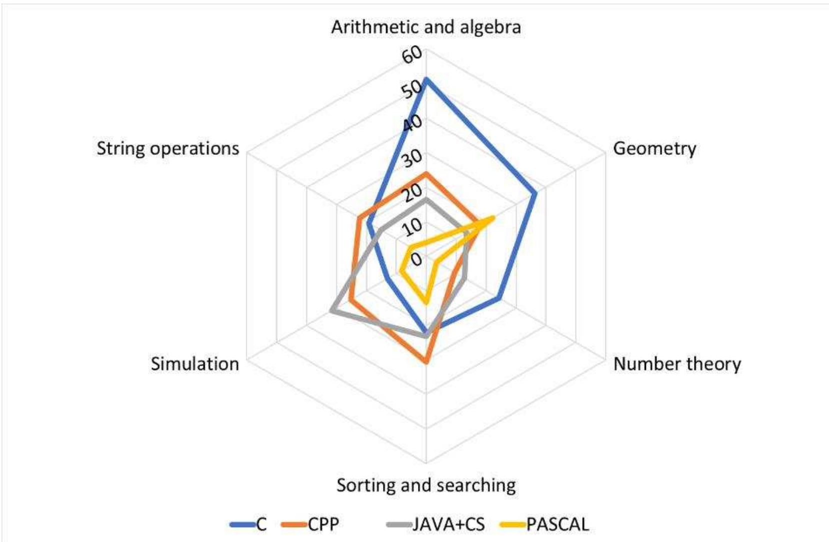


Figure 10

Distribution of submitted solutions by programming languages and task categories

In the left part of Figure 11, we compare the ratio of submitted solutions (outer ring) and accepted submissions (inner ring) for mathematical topics and do the same on the right for IT topics. It is clear that in both cases the C programming language leads in terms of the number of submissions. It is quite striking that in the case of tasks that also require mathematical skills, the acceptance rate of solutions in C programming language is remarkably high: 0.3689 (912 out of 2 472). In this respect, the IT tasks solved in C++ programming language are in second place with only 0.25 (107 out of 428).

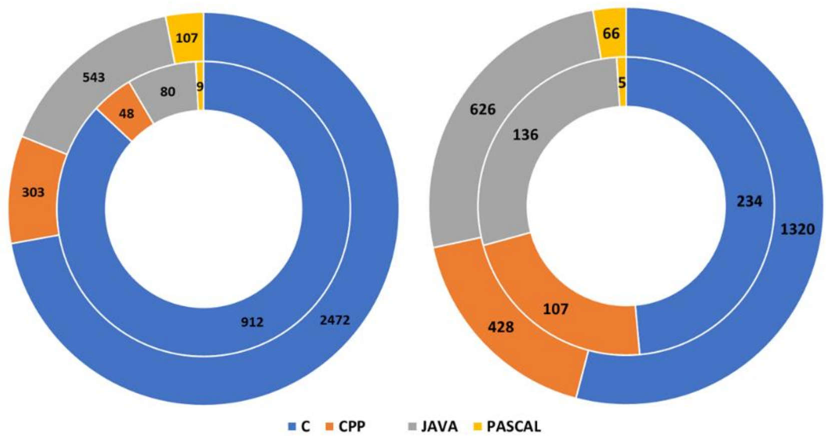


Figure 11
The ratio of total and successful submissions by programming language

Conclusions

In our article, we performed an analysis of a well-defined group of automatically evaluated programming tasks using the ProgCont system of the University of Debrecen, looking for the relationship between the performance of tasks requiring mathematical and IT knowledge. From the more than 277 000 automatically evaluated submitted solutions, we have selected the nearly 6 000 that belonged to the properly categorized tasks. We had previously set up 3 hypotheses, which were tested.

We found that, contrary to our expectations, users perform better on tasks that require proficiency in both disciplines. Therefore, it seems worthwhile to place more emphasis on tasks that require math skills in the teaching of programming, which also develops students' mathability skills [1]-[10], [18]. At the same time, it is worth rethinking the choice of the first programming language taught (as is currently the case in some of our majors).

As a result of our research, we have identified those task types (within the examined task types) where the performance of the users is weaker. At the same time, we set the direction for expanding the set of tasks that form the basis of the research.

We identified our users' preferences for the choice of programming language for the two large categories of the examined tasks. We have seen that the choice is not the best right now. This needs to be corrected during education.

Acknowledgement

This work was supported by the construction EFOP-3.6.3-VEKOP-16-2017-00002. The project was supported by the European Union, co-financed by the European Social Fund.

References

[1] Baranyi, P., Csapó A.: Definition and Synergies of Cognitive Infocommunications, Acta Polytechnica Hungarica, 2012, 9, 67-83

[2] Baranyi, P., Gilanyi, A.: Mathability: Emulating and Enhancing Human Mathematical Capabilities, 4th IEEE International Conference on Cognitive Infocommunications, 2013, 555-558

[3] Baranyi, P., Csapó, A., Várlaki, P.: An Overview of Research Trends in CogInfoCom. In: Szakál A (ed.) 18th International Conference on Intelligent Engineering Systems - INES 2014, Tihany: IEEE Hungary Section, 2014, pp. 181-186

[4] Biró, P., Csernoch, M., Abari, K., Máth, J.: First Year Students' Algorithmic Skills in Tertiary Computer Science Education. In: Kunifuji, S., Papadopoulos, G., Skulimowski, A., Kacprzyk, J. (eds) Knowledge, Information and Creativity Support Systems. Advances in Intelligent Systems and Computing, Vol. 416, Springer, Cham., 2016, https://doi.org/10.1007/978-3-319-27478-2_24

[5] Biró, P., Csernoch, M.: The mathability of computer problem solving approaches, In 2015 6th IEEE International Conference on Cognitive Infocommunications (CogInfoCom2015) (pp. 111-114), http://doi.org/10.1109/CogInfoCom.2015.7390574

[6] Biró, P., Csernoch, M.: The mathability of spreadsheet tools, In 2015 6th IEEE International Conference on Cognitive Infocommunications (CogInfoCom2015) (pp. 105-110), http://doi.org/10.1109/CogInfoCom.2015.7390573

[7] Chmielewska, K., Gilányi, A., Łukasiewicz, A.: Mathability and Mathematical Cognition, in 7th IEEE Conference on Cognitive Infocommunications (CogInfoCom) IEEE, 2016, 245-250

[8] Chmielewska, K., Gilányi, A.: Educational context of mathability. Acta Polytechnica Hungarica 15 (5), 223-237

[9] Chmielewska, K., Gilányi, A.: Mathability and Computer-aided Mathematical Education, in 6th IEEE Conference on Cognitive Infocommunications (CogInfoCom) IEEE, 2015, 473-477

[10] Chmielewska, K., Matuszak, D.: Mathability and Coaching, in 8th IEEE Conference on Cognitive Infocommunications (CogInfoCom) IEEE, 2017, 427-431

[11] Falus, I.: Introduction to the methodology of pedagogical research. (Bevezetés a pedagógiai kutatás módszereibe) Műszaki Kiadó, Budapest, 2004

[12] Futschek, G.: Algorithmic Thinking: The Key for Understanding Computer Science. Informatics Education–The Bridge between Using and Understanding Computers, 4226, 2006, 159-168

[13] Kádek, T., Biró, P.: On the way to the study group application with the ProgCont API. (Úton a szakköralkalmazás felé a ProgCont API-val.) INFODIDAKT 2019 konferencia, Zamárdi, Webdidaktika Alapítvány, https://people.inf.elte.hu/szlavi/InfoDidact19/Manuscripts/KTBP.pdf

[14] Kádek, T., Biró, P.: The ProgCont API: innovative evaluation of solutions to programming assignments. (A ProgCont API: programozási feladatok megoldásainak újszerű kiértékelése.) SZÁMOKT 2019, Temesvár, Románia: Erdélyi Magyar Műszaki Tudományos Társaság (EMT) kiadó, 2019, 191-195

[15] Kádek, T., Biró, P.: Effects of distance education on the ProgCont system. (A távolléti oktatás hatásai a ProgCont rendszerre.) In: ENELKO SzámOkt 2020, Erdélyi Magyar Műszaki Tudományos Társaság EMT, Kolozsvár, 2020, 104-109

[16] Kádek, T., Kósa, M., Pánovics, J.: Informatics competition tasks. (Informatikai versenyfeladatok.) Gyires Béla Informatikai Tananyag Tárház, 2014, https://gyires.inf.unideb.hu/GyBITT/04/

[17] Kósa, M., Pánovics, J., Gunda, L.: An Evaluating Tool for Programming Contests. Teaching Mathematics and Computer Science, 3/1, 2005, 103-119

[18] Kővári, A., Rajcsányi-Molnár, M.: Mathability and Creative Problem Solving in the MaTech Math Competition. Acta Polytechnica Hungarica, 2020, 17, 147-161

[19] Pólya, G.: How to solve it, United States of America: Princeton University Press, 1957

[20] Wing, J. M.: Computational thinking. Communications of the ACM, 49(3), 2006, 33-35, https://doi.org/10.1145/1118178.1118215

Measuring the Algorithmic Skills of Students Working with Low- and High-Mathability Programming Approaches

Katalin Sebestyén[a], Gábor Csapó[a], Mária Csernoch[b], Kálmán Abari[c]

[a]University of Debrecen, Doctoral School of Informatics
Kassai út 26, H-4028 Debrecen, Hungary
{sebestyén.katalin, csapo.gabor}@inf.unideb.hu

[b]University of Debrecen, Faculty of Informatics
Kassai út 26, H-4028 Debrecen, Hungary
csernoch.maria@inf.unideb.hu

[c]University of Debrecen, Faculty of Arts and Humanities
Egyetem tér 1, H-4032 Debrecen, Hungary
abari.kalman@arts.unideb.hu

Abstract: In spite of the fact that teaching programming is obligatory in Hungarian public ICT education, the low number of lessons accompanied by the lack of students' knowledge and interest marginalizes the topic. Research and experience show that even when working with experienced teachers, students have a hard time mastering imperative and object-oriented programming languages. These languages usually approach problem-solving from a mathematical perspective, with a minimal design IDE (Integrated Development Environment) and output screen. As an alternative solution, schools and courses apply various visual programming environments that make it possible to create colorful motivating games and animations even in one lesson. In our research, we compared two visual programming environments: Scratch (control group), developed for education, and Construct 3 (experimental group), developed for game and software development. We conducted measurement of the efficacy of the two environments for teaching programming in two grade-8 groups. The students learned the topic by solving traditional algorithmic tasks but taking advantage of the visual interfaces. The results, in accordance with previous findings, show that in developing the students' algorithmic skills there is no difference between these visual programming languages. Furthermore, we found proof that the selected teaching methods play a crucial role in the development of said skills of the students.

Keywords: visual programming; algorithmic skills; Bebras; ICT education; programming education; Construct 3; Scratch

1 Introduction

The process of developing students' computational thinking and algorithmic skills is restricted to the programming topic of ICT (Information and Communications Technology) education, according to the Hungarian Base Curricula [1] and Frame Curricula [2]. The low number of lessons, the ambiguous requirements of the Frame Curricula [3], and the outdated programming knowledge of ICT teachers marginalize the programming topic. This not only means that programming receives much less emphasis on the subject than was originally intended, but in several cases the topic is completely ignored or the focus is strictly on the tools, such as Scratch and robots instead of programming.

To teach programming in ICT education, several programming approaches and environments are widely accepted. The modern object-oriented languages (for example C++, C#, Python and Java) are present, but students can also encounter procedural languages which are outdated and are rarely used in the industry (like Pascal). Learning programming is difficult and challenging for beginners, considering both the problem-solving aspect and the syntactical rules of these high-level programming languages [3] [5] [6]. To make programming education more effective and more easily understandable for students, several educational programming languages (EPLs) have been introduced and have become widely accepted in the meantime. These languages usually take a different approach to create the code compared to text-based languages. [7] [8] [9] [10] [11] [12]. However, despite these attempts, students at the end of their secondary education do not possess the required level of algorithmic skills even for solving simple problems [12].

2 Visual Programming

In visual programming, students use pre-defined graphical language elements to construct the code of the program. This process emphasizes the building of algorithms without the burden of syntactical rules. The widespread use of this programming method in the industry is supported by several programming environments and game engines alongside the text-based options [14] [15] [16]. The visual representation of codes and the rapid developmental experience make visual programming languages compelling choices for the educational field, as well [17] [18].

It is worth noting that the various visual languages are not compatible with each other: a code created in one environment cannot be transferred directly to other environments. Analyzing the visual programming languages present in education and industry, we can define four categories based on their concepts [19] [20]:

- behavior-based,

- event-action based,
- block-based,
- node-based.

2.1 Scratch

The well-known Scratch, block-based visual programming environment is present at all levels of ICT education [11] [21]. It is primarily designed for beginner programmers, especially for students who could not imagine themselves as programmers before working with Scratch [11]. The environment includes several components designed for education (for example the ability to share projects or the public availability of project source codes). Despite the fact that students find this environment easy to use, several studies have encountered problems in terms of its effectiveness. Testing grade-5 students in a primary school, Kalelioglu & Gülbahar found that focusing on the spatial-visual aspect of Scratch did not increase problem-solving skills compared to traditional approaches [22]. Students tend to develop and follow bad programming habits while working with Scratch. For example, they include all blocks in a program code that might be needed or might be connected to the problem, without analyzing the original problem and the tools available. Additionally, students tend to over-deconstruct problems without logical coherence between the elements [23]. It is also important to emphasize that Scratch does not reinitialize variables upon re-executing a project. This leads to bad initialization habits and makes knowledge-transfer to further programming environments more difficult [24].

2.2 Construct 3

Our research group selected the HTML5 based general-purpose Construct 3 environment [25] which uses an implementation of the event-action-based and behavior-based visual programming approaches. With Construct 3, students can create simple 2D games and multimedia web applications quickly and easily. Consequently, the environment fulfills all the requirements of teaching fundamental programming concepts.

Construct 3 provides a complete graphical interface for creating projects. The users work with objects that are placed on layouts representing how the application will look visually on screen. These objects are pre-made elements that cover different functionalities of a project (like displaying a sprite or playing a sound). The environment includes various pre-programmed behaviors (for example different movement algorithms) which can be attached to objects. For creating custom logic, the users can use event-sheets to build up the visual code connecting actions to events referencing the objects. Further details on the workflow of the Construct environment are presented in our previous paper [19].

Construct 3 is well-documented, which aids its integration into classes [26] [27] [28] [29] [30]. To further support the education processes, the environment provides an option, similarly to Scratch, to publicly share students' work online. The free version of the environment can only be used with limitations. However, the capabilities of the free version are sufficient to teach programming in elementary and secondary ICT education.

2.3 Helping Materials and Tasks

Similar to other topics in ICT education [31] [32] [33], programming is also exposed to erroneous tasks, algorithms, and source codes that are built up without deeper understanding or logic. In general, low-mathability approaches are applied in the teaching-learning process, where the focus is on the tools instead of the problem [34].

Even the environments designed for educational contexts are burdened with such tasks. Figure 1 presents the source code of a Scratch task [34] which writes out the numbers divisible by five in the range of 100-150. The source code is based on several unusual solutions: for no apparent reason, the program only runs when the user presses the "o" character, and it uses an infinite loop (forever command) to write out the numbers which is only stopped by a separate if statement. This could be interpreted as a do-while loop; however, the task description in [34] includes no information on this knowledge item whatsoever.

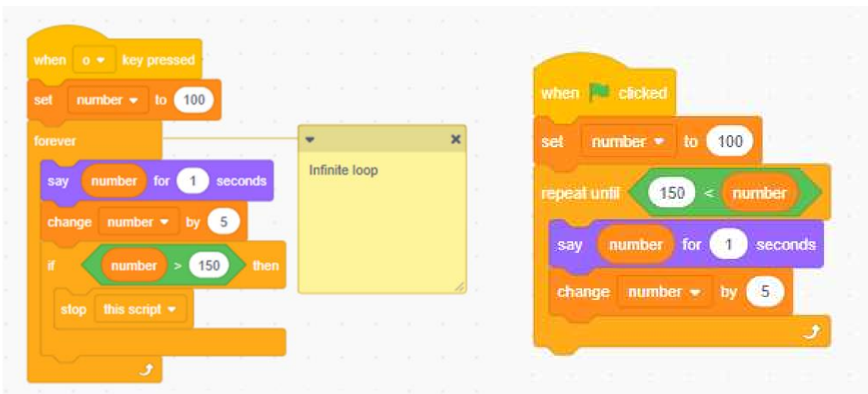


Figure 1
An example task in the Scratch environment for writing outnumbers divisible by five in the range 100–150 [34] (left) The modified version of the original source code (right) (The source codes are the English translation of the original Hungarian version)

We selected this task to highlight the erroneous problem-solving strategies present in ICT education. Note that the task presented and the errors included in it do not represent the whole world of ICT, as one can find educational materials and tasks of outstanding quality, just as there are ICT teachers who are accurate and professional (expert teachers) **Error! Reference source not found.** [36]. In general, we can conclude that it is time to rethink, correct, and transform ineffective practices, and that this is in the interest of all parties involved in education.

2.4 Our Goals

In this paper, we present an analysis of the effectiveness of the Scratch and Construct 3 environments' development of students' algorithmic skills. Our goal was to examine how different visual programming approaches affect the development of said skills by solving tasks with methods that focus on problem-solving – high mathability [38] – instead of on the graphical interfaces and visual capabilities of the environments – low mathability [34] [38] [39] [41].

3 Applied Methods, Tools, and Strategies

3.1 Research Environment

We analyzed the effectiveness of the methods used in programming education in two grade-8 groups in a local high school (experimental and control groups). The students took part in the 6-year training program of the school and during their previous education, they had not encountered the programming topic. Both groups progressed and learned the knowledge items at the same pace. The experimental group used Construct 3, while the control group learned with the Scratch environment. During the teaching period, we aimed to minimalize the differences between the two environments by developing customized tasks. Therefore, both groups followed the same schedule and worked on the same tasks optimized for their environment. The experimental group studied the topic for 18 lessons, while the control group for 17 lessons, one lesson per week.

3.2 Applied Methods

The main roles of the first tasks were that the students could finish simple projects and get feedback on their work, and additionally, learn the fundamentals of creating projects in their environment. After the groups grasped the essence of the interface, and it did not hinder the real problem-solving process, the complexity of the tasks was increased.

To solve the tasks, the students were guided by the coaching method [42], based on Pólya's [42] concept-based problem-solving approach:

1) Presenting the task.

2) Understanding the problem: Analyzing the complete task and decomposing it into subtasks.
3) Setting the goal of the subtasks. Highlighting the input and output values of the subtasks.
4) Building the algorithm.
5) Precoding in natural language: While following the structure of the visual programming language, phrasing the conditions and statements required to code.
6) Coding.
7) Testing: Running the code and discussing the outputs.
8) Debugging and correcting the errors.
9) Combining the subtasks: debugging, discussion, abstraction.

Note, that designing tasks for visual programming environments involves additional factors that teachers must take into consideration. Besides the creation of the project and the algorithms included in it, the visual design of the project also plays an important role in the success of each task. During our work, besides minimalizing the differences originating from the two environments, we also focused on avoiding error prone approaches). The teaching of both the experimental and control groups was carried out by the same teacher from our research group.

The problems, the students worked on were presented in visually engaging smaller projects. At the beginning of these projects, the teacher presented the complete work and started a discussion following the aforementioned approach. The projects were then decomposed into smaller subtasks which the groups analyzed further, before starting the development process. For example, in one of the projects the groups created a game with randomly appearing targets and a sling that could shoot projectiles. The students analyzed the problem both from the point of the required objects and assets and from the point of the algorithms behind the behavior of these elements. During the teaching period, the following projects were created:

- crossing the street (simple movement, collision detection, handling variables, outputting text referencing variables, and random number generation),
- target practice (handling user input, creating advanced logic),
- UFO attack (exercise task).

3.3 Data Collection

We collected data in pre- and post-tests, using paper-based test sheets. The pre-test was carried out before the first programming lesson to avoid affecting the prior knowledge of the students. The post-test was administered during the lesson following the last programming class.

Table 1
The number of students in the experimental and control groups

	experimental group	control group
number of students	14	14
pre-test	10	13
post-test	12	13
paired tests	7	12

Based on the Frame Curricula and the local curricula of the school, the programming topic only appears in grade-8 and grade-10 classes. In the school year of the measurement, only one grade-8 class was enrolled, and the class was divided into two groups.

The low number of students in both groups made it possible to ensure a similar progression in the topic (Table 1). The fluctuation in the number of students (especially considering the paired tests in the experimental group) can be explained by the students' absences and various school activities.

3.4 Tests

To measure the algorithmic skills of the students, we composed a test that relies on Bebras tasks [44] [45], making it independent of the programming environment. We analyzed and selected the Bebras tasks we included to make them cover several aspects of the algorithmic skills, requiring differing thinking processes and strategies to solve. The selected tasks were not focused directly on the knowledge items present in the problems solved. Instead, our goal was to measure the algorithmic and problem-solving skills developed during the teaching-learning period. It is important to note that we found differences between the original tasks [46] and the Hungarian versions we used [44]. However, these differences are minimal and have no effect on the skills measured. We changed several multiple-choice tasks to open-answer tasks so that students did not have predefined options from which they could select one randomly if they could not solve the problem. Following these conditions, the testing process included these altered tasks (Appendix, section 0).

3.5 Analyzing the Data

To store the data collected with the tests, we created a database where the tasks were decomposed into items. Due to the particularities of the tasks, the student answers were on a narrow scale as several tasks required a character or number as the answer, despite the open questions. Therefore, the decomposition of the students' answers into items was limited.

POPULARITY (2015-CA-01)

Seven beavers are in an online social network called Instadam. Instadam only allows them to see the photos on their own and their friends' page. In this diagram, if two beavers are friends they are joined by a line. After the supper holiday everybody posts a picture of themselves on all of their friends' page.

Which beavers' picture will be seen the most?

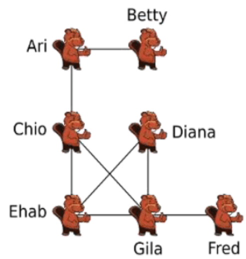


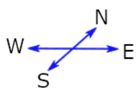
Figure 2
The popularity task (2015-CA-01) [45] [46]

During the analysis of the data, we stored additional information on the students' answers where the tasks justified it. The *popularity* task is an example of this, where a frequently chosen answer was the character who seemingly has the most connections (Figure 2).

In this case, we not only marked the answer incorrect but grouped it into a separate category to make it possible to observe the differences in how students applied fast-thinking [47]. Similarly, in the *spherical robot* task (Figure 3), we also separated those answers which guided the ball into the goal with unnecessary instructions from those that completed the task perfectly.

SPHERICAL ROBOT (2016-JP-03)

The BeaverBall is a toy that can be operated by remote control, and understands each of four direction commands.



If the BeaverBall moves to a white square, it drops down one level. The BeaverBall ignores commands that cause it to move outside the borders.

Look at the position of the BeaverBall in the picture above. Which of the following lists of directions will cause the BeaverBall to reach the GOAL?

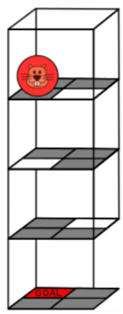


Figure 3
The spherical robot task (2016-JP-03) [44] [46]

4 Analyzing the Effectiveness of Programming Tools

Table 1
The results (%) of the pre- and post-tests in the experimental (exp.) and control groups by task, along with the p-values of the differences between the groups

Tasks	**Pre-test**		**p**	**Post-test**		**p**
	exp.	**control**		**exp.**	**control**	
rotating puzzle	90.00	69.23	0.226	83.33	84.62	0.934
popularity	70.00	38.46	0.030	70.83	53.85	0.059
beaver code	100.00	100.00		100.00	100.00	
party guests	70.00	63.08	0.718	83.33	87.69	0.752
hierarchy	80.00	84.62	0.784	91.67	100.00	0.339
spherical robot	50.00	76.92	0.195	91.67	84.62	0.606
deactivatin	80.00	23.08	0.005	66.67	69.26	0.896
concurrent directions	80.00	53.85	0.197	91.67	86.92	0.329
four errands	65.00	84.62	0.254	100.00	86.54	0.047
kix code	22.50	42.31	0.292	89.58	65.38	0.130
blossom	46.00	81.54	0.074	91.67	98.46	0.413
total	59.61	65.97	0.429	88.46	83.13	0.396

To measure the effectiveness of the two programming interfaces (experimental group: Construct 3, control group: Scratch) in developing the algorithmic skills of students, we analyzed the results of the pre- and post-tests (Table 1). At this step of the research, we included the results of all students, regardless of whether they were present at both the pre- and the post-tests. For analyzing the data and the difference between the groups, we used the SPSS software package [48].

Considering the results of the pre-test, both groups provided a high proportion of correct answers. This meant that no significant differences could be found between the experimental and control groups ($p = 0.429$), except for two tasks: *popularity* ($p = 0.030$) and *deactivatin* ($p = 0.005$).

Popularity: The experimental group completed the task more successfully, with 70%, while the control group achieved 38.46%. A high proportion of students marked the incorrect option, i.e. "Gila", who has the most direct connections. This implies that they answered the task by relying on their first impression without further analyzing the problem in hyper-attention mode [49] [50] (Figure 2).

Deactivatin: Students giving incorrect answers mostly marked jars D or E, which can be explained by a shallow interpretation of the task, again applying the hyper-attention mode [49] [50] (Figure 4).

DEACTIVATIN (2016-HU-06)

Beavers Carl and Judy have 3 jars of candies each.

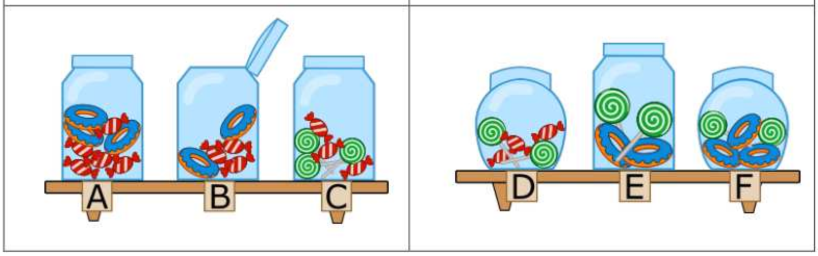
Each jar can have several of the following 10 properties:

- It's either open (1) or closed (2).
- It either contains red candies with white stripes (3) or not (4).
- It either contains blue sugar donuts (5) or not (6).
- It either contains green spiral lollypops (7) or not (8).
- Its form is either round (9) or angular (10).

For example Carl's "A" jar has the following properties: 2, 3, 5, 8 and 10.

You can notice that Carl's jars have got some common properties just like Judy's jars.

Carl's jars	Judy's jars



Which jar has both the properties that are common among Carl's jars and Judy's jars separately?

Figure 4
The deactivatin task (2016-HU-06) [44] [46]

Considering the results of the post-test, the groups completed the test with similar rates of success, without significant differences (p = 0.396). Analyzing the tasks, only the *four errands* task shows a significant difference between the experimental and control groups (p = 0.047).

Examining the results of both the pre- and post-tests, both groups show significant development in the topic (experimental group: p = 0.002; control group: p = 0.022). Furthermore, in the post-test the experimental and control groups showed no significant difference in solving the Bebras tasks [44] [46]. This allows us to conclude that the development of the students' algorithmic skills is independent of the visual programming environment.

Conclusion

In this paper, we presented research on how an algorithm-driven teaching method in different visual programming environments and languages develops the students' algorithmic skills, and what differences can be observed between the groups. The sample included two grade-8 student groups who studied the programming topic at a similar pace. The experimental group studied with Construct 3, while the control group with Scratch. Both groups had the same teacher and worked on the same programming problems using a concept-based [42], coaching method [42].

The data collection was carried out applying printed test sheets composed of Bebras tasks. Considering the analysis of the data (Table 1) both groups completed both tests at similar levels, without significant differences.

Based on the results, the Construct 3 environment created for software production does not develop the students' algorithmic skills more effectively than the Scratch environment designed for educational purposes. However, both environments develop the algorithmic skills of students significantly when the tasks focus on problem-solving, instead of on the graphical, drawing functions of the programs. The results show that while the used software environment does not have an effect on the development of the students' algorithmic skills, the method and/or the teaching approach applied to **Error! Reference source not found.**.

Acknowledgements

This work was supported by the construction EFOP-3.6.3-VEKOP-16-2017-00002. The project was supported by the European Union, co-financed by the European Social Fund.

References

[1] NAT 2012: 110/2012 (VI. 4.) Government regulation on the publication, introduction and application of the Base Curricula, 2012, In Hungarian: Nemzeti Alaptanterv, 2012 [Online] Available: http://ofi.hu/sites/default/files/attachments/mk_nat_20121.pdf [Accessed: 04-Nov-2019]

[2] OFI: Frame Curricula. In Hungarian: Kerettanterv. 51/2012. (XII. 21.) számú EMMI rendelet – a kerettantervek kiadásának és jóváhagyásának rendjéről, 2012 [Online] Available: https://www.oktatas.hu/kozneveles/kerettantervek/2012_nat [Accessed: 04-Febr-2020]

[3] Nagy T. K.: The paradox of the hungarian frame curricula in informatics. The Turkish Online Journal of Educational Technology, INTE 2018, pp. 910-922

[4] Booth, S.: Learning to program: A phenomenographic perspective. Acta Universitatis Gothoburgensis, Gothenburg, Sweden, 1992

[5] Soloway, E.: Should we teach students to program? Communications of the ACM, 1993, 36(10), pp. 21-25

[6] Ben-Ari, M. (2011). Non-myths about programming. Communications of the ACM, 2011, 54(7), pp. 35-37

[7] Fincher, S. et al.: Comparing Alice, Greenfoot & Scratch. 41st ACM technical symposium on Computer science education, ACM, 2010, pp. 192-193

[8] Fowler, A., Fristce, T. & MacLauren, M.: Kodu Game Lab: a programming environment. The Computer Games Journal, 2012, 1(1), pp. 17-28

[9] Klassner, F. & Anderson, S. D.: Lego MindStorms: Not just for K-12 anymore. IEEE Robotics & Automation Magazine, 2003, 10(2), pp. 12-18

[10] Papadakis, S. et al.: Novice Programming Environments. Scratch & App Inventor: a first comparison. Workshop on Interaction Design in Educational Environments, ACM, 2014

[11] Resnick, M. et al.: Scratch: programming for all. Communications of the ACM, 2009, 52(11), pp. 60-67

[12] Chmielewska, K. & Gilanyi, A.: Mathability and computer aided mathematical education. 6th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), 2015, pp. 473-477

[13] Csernoch, M., Biró, P., Máth, J. & Abari, K.: Testing Algorithmic Skills in Traditional and Non-Traditional Programming Environments. Informatics in Education, 2015, 14(2), pp. 175-197

[14] "Blueprints Visual Scripting" Epic Games. [Online] Available: https://docs.unrealengine.com/en-US/Engine/Blueprints/index.html [Accessed: 10-Oct-2019]

[15] Hutong Games: PlayMaker – Visual Scripting for Unit. [Online] Available: https://hutonggames.com/ [Accessed: 10-Jan-2020]

[16] YoYo Games: Drag And Drop Overview [Online] Available: https://docs2.yoyogames.com/ [Accessed: 10-Jan-2020]

[17] Fesakis, G. & Serafeim, K.: Influence of the familiarization with scratch on future teachers' opinions and attitudes about programming and ICT in education. ACM SIGCSE Bulletin, ACM, 2009, 41(3), pp. 258-262

[18] Kim, H., Choi, H., Han, J. & So, H. J.: Enhancing teachers' ICT capacity for the 21st Century learning environment: Three cases of teacher education in Korea. Australasian Journal of Educational Technology. 2012, 28(6) pp. 965-982

[19] Csapó, G.: Placing event-action based visual programming in the process of computer science education. Acta Polytechnica Hungarica, 2019, 16(2) pp. 35-57

[20] Csapó, G. & Sebestyén, K.: Educational Software for the Sprego Method. The Turkish Online Journal of Educational Technology, INTE 2017 October, pp. 986-999

[21] Lifelong Kindergarten Group:Scratch – Imagine, Program, Share [Online] Available: https://scratch.mit.edu [Accessed: 10-Nov-2019]

[22] Kalelioglu, F. & Gülbahar, Y.: The effects of teaching programming via Scratch on problem solving skills: a discussion from learners' perspective. Informatics in Education, 2014, 13(1), pp. 33-50

[23] Meerbaum-Salant, O., Armoni, M. & Ben-Ari, M.: Habits of programming in scratch. 16th annual joint conference on Innovation and technology in computer science education. ACM, 2011, pp. 168-172

[24] Franklin, D., Hill, C., Dwyer, H. A., Hansen, A. K., Iveland, A. & Harlow, D. B.: Initialization in scratch: Seeking knowledge transfer. 47th ACM Technical Symposium on Computing Science Education, ACM, 2016, pp. 217-222

[25] "Game Making Software – Construct 3" [Online] Available: https://www.construct.net/en [Accessed: 21-Sep-2019]

[26] Alexander, J.: Construct 2 – From Beginner to Advanced - Ultimate Course! | Udemy. 2016 [Online] Available: https://www.udemy.com/construct-2-from-beginner-to-advanced-build-10-games [Accessed: 06-Dec-2019]

[27] Scirra: Welcome to the Construct 3 Manual. [Online] Available: https://www.construct.net/en/make-games/manuals/construct-3 [Accessed: 21-Sep-2019]

[28] Scirra: Game Development Tutorials. [Online] Available: https://www.construct.net/en/tutorials?flang=1 [Accessed: 21-Oct-2019]

[29] Scirra: Construct Forum [Online] Available: https://www.construct.net/en/forum [Accessed: 21-Sep-2019]

[30] Scirra: ScirraVideos – YouTube [Online] Available: https://www.youtube.com/user/ScirraVideos [Accessed: 20-Nov-2019]

[31] Papp, P.: Texting or text management? Teaching text management in ICT textbooks. In Hungarian: Szövegelés vagy szövegszerkesztés? Szövegkezelés tanítása az informatika tankönyvekben. Debreceni Egyetem Informatikai Kar Tudományos Diákköri Konferencia 2019

[32] Erdélyi, A.: Down the yellow paved road of ICT textbooks. In Hungarian: Végig az informatika-tankönyvek sárga köves útján – Avagy Pöttömföldről eljutunk-e így az igazi varázslathoz? Debreceni Egyetem Informatikai Kar Tudományos Diákköri Konferencia 2019

[33] Nagy, R. K.: Learning spreadsheet management from textbooks: do we only manage the spreadsheets or do we solve problems? In Hungarian: Táblázatkezelés elsajátítása tankönyvekből: a táblázatot csak kezeljük vagy a problémát is megoldjuk? Debreceni Egyetem Informatikai Kar Tudományos Diákköri Konferencia 2019

[34] Baranyi, P. & Gilányi, A.: Mathability: Emulating and Enhancing Human Mathematical Capabilities, IEEE 4th International Conference on Cognitive Infocommunications (CogInfoCom), 2013, pp. 555-558

[35] Burcsi: Scratch programming. In Hungarian: Scratch programozás [Online] Available: http://www.burcsi.hu/_inf/Scratch/ [Accessed: 12-Nov-2019]

[36] Hattie, J.: Visible Learning for Teachers: Maximizing Impact on Learning, Routledge, 2012

[37] Csernoch, M.: Thinking Fast and Slow in Computer Problem Solving, Journal of Software Engineering and Applications, 2017, 10(1), pp. 11-40

[38] Chmielewska, K. & Gilányi, A.: Computer Assisted Activating Methods in Education. 10th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), 2019, pp. 241-246

[39] Biró, P. & Csernoch, M.: The mathability of computer problem solving approaches. 6th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), 2015, pp. 111-114

[40] Biró, P. & Csernoch, M.: The mathability of spreadsheet tools. 6th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), 2015, pp. 105-110

[41] Chmielewska, K., Gilányi, A. & Łukasiewicz, A.: Mathability and Mathematical Cognition. 7th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), 2016, pp. 245-250

[42] Chmielewska, K. & Matuszak, D.: Mathability and coaching. 8th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), 2017, pp. 427-432

[43] Pólya, G.: How To Solve It. A New Aspect of Mathematical Method. Second edition (1957) Princeton University Press, Princeton, New Jersey, 1954

[44] ELTE IK T@T Labor: Archivum | e-Hód. [Online] Available: http://e-hod.elte.hu/archivum/ [Accessed: 09-Aug-2019]

[45] Pohl, W. & Dagienė, V.: What is Bebras | www.bebras.org [Online] Available: https://www.bebras.org/ [Accessed: 15-Jan-2020]

[46] Bebras: International Challenge on Informatics and Computational Thinking [Online] Available: https://www.bebras.org [Accessed: 15-Jan-2020]

[47] Kahneman, D.: Thinking, Fast and Slow. Farrar, Straus and Giroux, New York, 2011

[48] IBM SPSS software: IBM SPSS Statistics [Online] Available: https://www.ibm.com/analytics/spss-statistics-software [Accessed: 01-Jan-2020]

[49] Dani, E.: The HY-DE Model: An Interdisciplinary Attempt to Deal with the Phenomenon of Hyperattention. Journal of Systemics, Cybernetics and Informatics 2016, 13:(6) pp. 8-14

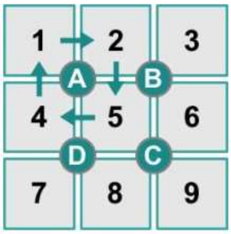
[50] Csernoch, M. & Dani, E.: Data-structure validator: an application of the HY-DE model, 8th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), 2017, pp. 197-202

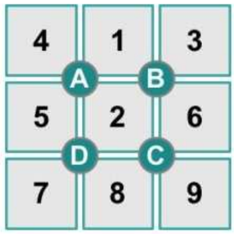
Appendix

1.1. The test sheet used for data-collection

ROTATING PUZZLE (2012-HU-01A)

Henry Beaver plays a new game. If he presses one of the buttons A, B, C or D in this game, numbers around the button will be rotated clockwise as shown on the pictures in left. The result of pressing the button A is shown to the picture in right. The initial state of the numbers is as seen on the left image.





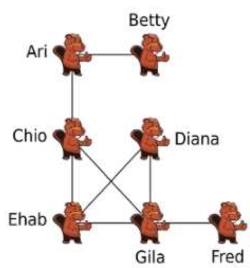
Henry Beaver pressed consecutively four buttons: D, C, B, B.

Where the number 4 happens to be after the pressing of the four buttons?

POPULARITY (2015-CA-01)

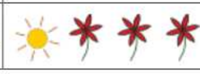
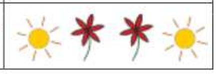
Seven beavers are in an online social network called Instadam. Instadam only allows them to see the photos on their own and their friends' page. In this diagram, if two beavers are friends they are joined by a line. After the supper holiday everybody posts a picture of themselves on all of their friends' page.

Which beavers' picture will be seen the most?



BEAVER CODE (2016-CH-12)

Barbara has been given two stamps. With one she can produce a little flower, with the other a little sun. Being a clever girl, she thinks of a way to write her own name by using the code.

Letter	B	A	R	E	Y
Code					

So Barbara becomes:

She then writes the name of her friend:

What is her friend's name?

PARTY GUESTS (2016-IS-02)

To arrange a dinner party Sara the beaver need to talk to five friends: Alicia, Beat, Caroline, David und Emil.

However, to talk to her other friends, there are a few points to consider:

1. Before she talks to David, she must first talk to Alicia.
2. Before she talks to Beat, she must first talk to Emil.
3. Before she talks to Caroline, she must first talk to Beat and David.
4. Before she talks to Alicia, she must first talk to Beat and Emil.

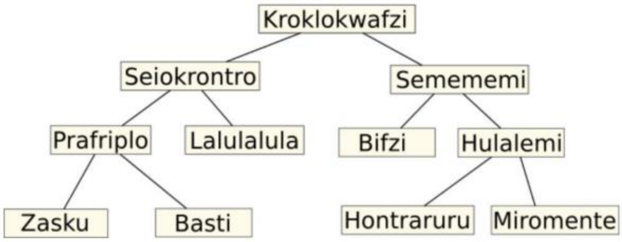
In what order should Sara talk to all of her friends if she wants to talk to all of them?

HIERARCHY (2016-CZ-03)

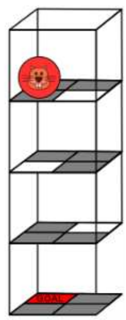
We have got a tree describing relationships between animals on Morgenstern planet. A link between two categories in the tree below means that all members of the lower category are also members of the upper category. Hence, some sentences can be formed looking at this tree, e. g. every "Hulalemi" is a type of "Semememi" and that some "Seiokrontro" are not a type of "Basti".

Only one of the following statements is true. Which one?

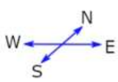
1. Every Basti is a Seiokrontro as well.
2. Some Hontraruru is not Semememi.
3. Every Zasku is Bifzi as well.
4. Every Prafriplo is Basti as well.



SPHERICAL ROBOT (2016-JP-03)



The BeaverBall is a toy that can be operated by remote control, and understands each of four direction commands.



If the BeaverBall moves to a white square, it drops down one level. The BeaverBall ignores commands that cause it to move outside the borders.

Look at the position of the BeaverBall in the picture above. With what commands can you make the BeaverBall to reach the GOAL?

DEACTIVATIN (2016-HU-06)

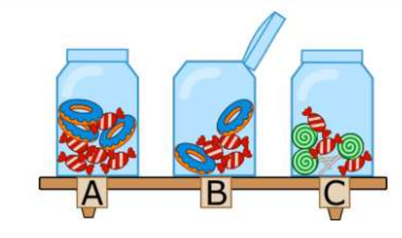
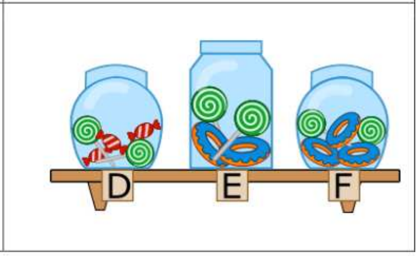
Beavers Carl and Judy have 3 jars of candies each.

Each jar can have several of the following 10 properties:

- It's either open (1) or closed (2).
- It either contains red candies with white stripes (3) or not (4).
- It either contains blue sugar donuts (5) or not (6).
- It either contains green spiral lollypops (7) or not (8).
- Its form is either round (9) or angular (10).

For example Carl's "A" jar has the following properties: 2, 3, 5, 8 and 10.

You can notice that Carl's jars have got some common properties just like Judy's jars.

Carl's jars	Judy's jars
	

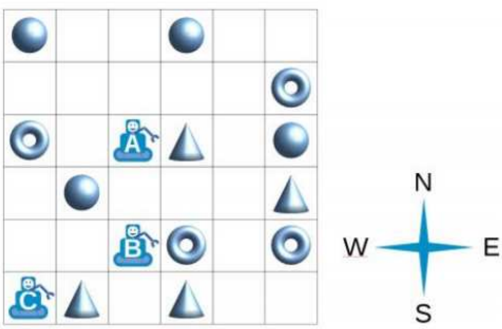
Which jar has both the properties that are common among Carl's jars and Judy's jars separately?

CONCURRENT DIRECTIONS (2016-IE-05)

In a warehouse, three robots always work as a team. When the team gets a direction instruction (N, S, E, W), all robots in the grid will move one square in that direction at the same time.

When the robots touch an item, they pick it up. You have to control them so that they will not pick up unnecessary items.

For example, if we give the list N, N, S, S, E to the team, then the robots will pick up two cones, and a ring.



What list of instructions can be sent to the robots so that the team picks up exactly a sphere, a cone, and a ring?

FOUR ERRANDS (2016-LT-03)

Beaver Alexandra wants to do the following tasks during her break (12:00 – 13:00):

- buy a book at a bookstore;
- buy a bottle of milk at a grocery;
- send the newly bought book by post;
- drink a cup of coffee in a cafeteria.

Alexandra estimated the time to complete each task. But these estimates are valid only outside of the busiest periods. So she is trying to avoid the busiest periods.

Place	Duration	Busiest periods
Bookstore	15 min	12:40 – 13:00
Grocery	10 min	12:00 – 12:40
Post office	15 min	12:00 – 12:30
Cafeteria	20 min	12:30 – 12:50

Help Alexandra order her tasks to make sure that she will avoid all of the busiest periods.

KIX CODE (2016-NL-04)

The Bebras Post Office uses postal codes that contain four characters. To make the postal codes readable by machines, they convert the postal codes into Kix codes.

In a Kix code, each character is represented by 4 vertical bars.

A code has 2 sections: upper and lower. The upper section contains only the middle and the top bars, while the lower section contains only the middle and the bottom bars. The middle bars overlap each other.

This table shows the codes for several characters:

0	1	2	3	4	5
6	7	8	9	A	B
C	D	E	F	G	H
I	J	K	L	M	N
O	P	Q	R	S	T
U	V	W	X	Y	Z

Example: The Kix code for "G7Y0" is

Question: Another postal code has this Kix code.

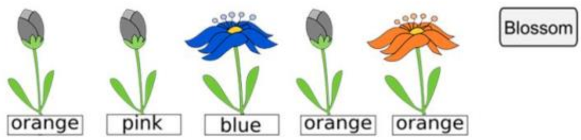
What is the postal code?

BLOSSOM (2016-SK-04)

Jane is playing a computer game. First the computer secretly chooses colours for five buds. The available colours for each flower are **blue**, **orange**, and **pink**. The chosen colours remain the same throughout the game. Jane has to guess which flower has which colour. She makes her first five guesses and presses the Blossom button.

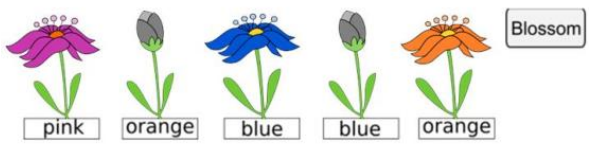
The buds, whose colours she guessed correctly, break into flowers. The others remain as buds.

Jane's first go:



Jane then has another go at guessing and presses the Blossom button again.

Jane's second go:



What colours did the computer choose for the flowers?

Do You Speak, Write and Think in Informatics?

Mária Csernoch, Erzsébet Dani

University of Debrecen, Faculty of Informatics, Kassai út 26, 4028 Debrecen, Hungary, csernoch.maria@inf.unideb.hu

University of Debrecen, Faculty of Humanities, Egyetem tér 1, 4032 Debrecen, Hungary, dani.erzsebet@arts.unideb.hu

Abstract: The dissemination of research results might be as crucial as the research itself. The widely accepted two major forms of dissemination are written papers and live presentations. On the surface, if we see the problem in hyper attention mode, these documents are different in nature. However, their preparation requires the same problem-solving approach, and beyond that, they share the fundamental rules of design since both are extended text documents, with varying proportions of text and/or non-text contents and static and/or dynamic media types. Closely related to this problem is the phenomenon of different types of attention (hyper and deep attention), thinking (fast and slow) modes, and problem-solving approaches (high- and low-mathability). In the world of immense and various forms of data and information, the role of so-called hyper attention is fundamental and inevitable, but the presence of deep attention is essential as well. In the present paper the knowledge items involved and shared in the design and the preparation of text-based documents are detailed from the view of concept-based problem-solving, the perspective of attention and thinking modes, along with samples originated from various sources and subject matters. The aims are to discuss the theoretical background of effective and efficient document design and preparation and to call attention to the consequences of ignoring, neglecting the proper use of attention types and thinking modes.

Keywords: end-user computing; attention modes; thinking modes; deep and surface computer problem-solving approaches; low- and high-mathability

1 Introduction

1.1 Who is an End-User and what Problems He/She is Faced with?

One of the most common scenarios is when an end-user is clicking around in computer applications suffering, but still claiming that – "I can do it, that's enough." – "No, that's not," – is the short reflection to this frequently repeated and widely accepted but low-mathability [1] [2] [3] slogan. The long explanation should cover all the efficient and effective problem-solving approaches accepted and practiced in other sciences, but hardly recognized and applied in end-user informatics and problem-solving.

By definition, “An end-user is the person that a software program or hardware device is designed for. The term is based on the idea that the ‘end goal’ of a software or hardware product is to be useful to the consumer. To simplify, the end-user is the person who uses the software or hardware after it has been fully developed, marketed, and installed” [4], in short, “the ultimate consumer of a finished product” [5].

However, problems arise when the complexity, the requirements, the set-up of a product is far beyond the scope of the target population. Further difficulties are faced when not one but a whole set of products are required to be handled. The figure becomes even more frightening when long forgotten or never learned knowledge, skills, abilities, attitudes[1] should be brought into these computational environments which effectiveness is clearly proved in high-mathability studies [8] [9] [14]. In general, when the right proportion of fast and slow thinking [6] should be activated for effective and efficient problem-solving, leaving low-mathability [3] methods behind. In the context of the attention issue: difficulties arise when the end-user needs to switch from hyper to deep attention [7].

1.2 The Real End-User

By experience, an end-user is the person who has never been educated in informatics/computer sciences (I/CS) or was taught how to navigate interfaces or self-taught or an I/CS professional but neglects or ignores non-professional informatics. A person who can be persuaded by software companies that they are in continuous need of newer versions and they do not need to know anything for using these applications, so they are the perfect ignorant targets.

This issue goes even further when experts are aware of end-users’ erroneous spreadsheet documents causing serious financial losses and search for explanations. In his research, Panko concluded that the behavior of end-users can be explained by their bad habit of thinking [10]-[12]. However, our research in computer problem-solving approaches, attention modes [7] [14], thinking modes [6], and mathability reveal that thinking is not bad, but applied in incorrect, non-suitable situations, primarily focusing on low-mathability approaches in hyper attention mode.

[1] Closely related to this issue is the topic of digital literacy and competence, because not even the new generations are born with digital skills which need to be consciously acquired and continuously developed, similar to mother tongue language skills.

1.3 The Problem-Solving Approaches of End-Users

The typology of computer problem-solving approaches [17] [18] and levels of understanding [19] [20], built upon traditional sciences with amendments matching the peculiarities of computers, was created for classifying computer-related activities (Figure 1). This typology consists of two hypernyms – deep and surface approaches – and five hyponyms, two and three, respectively.

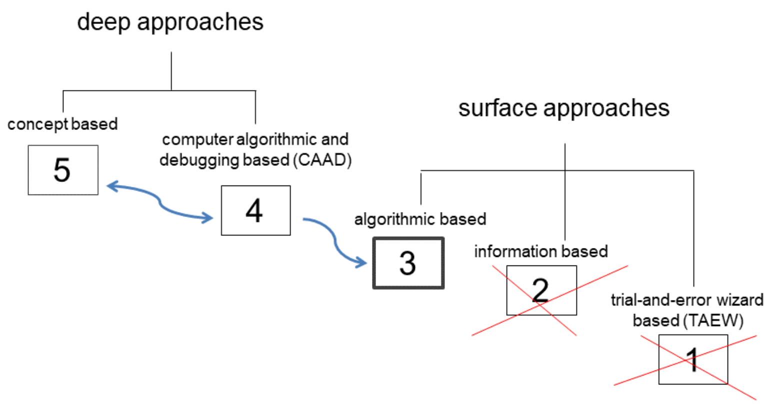


Figure 1

The typology of computer problem-solving approaches [17]

Considering the thinking modes – fast and slow [6] –, at Level 3 fast thinking, while at the other levels slow thinking work (Figure 1). With the arrows, the primary directions of communication between the different levels are indicated. There is continuous communication between Levels 5 and 4, where/when the first two steps of problem-solving take place. Level 4 is the position where schema-construction takes place, and these schemata are called/retrieved at Level 3. However, we must be aware that any backtracking is possible when errors or hiatus are recognized (these arrows are not included in Figure 1). Approaches at Levels 2 and 1 can hardly be considered problem-solving since they are strongly connected to the surfaces, they are interface navigation in hyper attention mode.

Considering the different computer problem-solving approaches and the documents handled – created and modified – by end-users clearly reveal that the experienced description of the end-user (Section 1.2) is more appropriate than the definition (Section 1.1). On the one hand, end-users are not able to function, as they should according to the definition. On the other hand, their productivity, effectiveness is much lower than excepted [10]-[12] [22]. Consequently, they perfectly fulfill the good customer image of software companies but lack computational thinking skills [21].

In the present paper, the speaking-writing-thinking approach of end-user computing is detailed, compared to the widely accepted surface approach, low-mathability erroneous methods (Figure 1). Furthermore, the essence of the concept-based problem-solving [23] – proved high-mathability methods [3] supported by the appropriate proportion of attention [6] and thinking modes [13] – is drawn, confronted with samples of erroneous practice [18] [22].

1.4 The Attention-Phase Perspective of the Typology

The problem raised in the present paper connects closely with the HY-DE (HYper and DEep attention) model at this point. The model was created by Erzsébet Dani [7] for the true consumers of the new technologies and applications, for the "bit-generations" (generations Y and Z). Since the basic unit of computer data is the bit, the children of the computer age can be identified with this denomination. The world of digital data produces the phenomenon of hyper attention, as described by Katherine N. Hayles [13]: "switching focuses rapidly among different tasks, preferring multiple information streams, seeking a high level of stimulation, and having a low tolerance for boredom". Based on these characteristics, without any proof, Prensky's concept of digital natives and immigrants indicates that there is no need for deep attention-driven activities in the digital era [24]. However, the pattern is not that simple –in contrast to Prensky [24] –, Kirschner & De Bruyckere [25] proved that digital native and multitasking is only a myth.

It must be noted, though, that surface or hyper-attention approaches have not only negative but also positive aspects. In fact, it is necessary for processing data in the world of the information superhighway. However, deep attention is also needed, because in this attention phase the focus is on one issue, which leads toward deeper, constructive thinking and the resolution of the recognized and understood problems [25].

The basic concept behind the model is the shifting pattern of the two forms of attention: hyper and deep attention. The main goal is to influence and control the hyper attention phase consciously in order to keep it in the background and effect a shift that activates the needed deep attention. The phase-change is connected by a mixed attention section, in which both attention states are present, but with reversed intensity (i.e., decreasing hyper attention and increasing deep attention).

The original theoretical model has two stages in the teaching-learning process: one for the instructor and one for the student. The forms of attention tend in opposite directions. In the instructor stage, attention-training moves from hyper to deep attention, with instructional dominance, while in the independent-student-activity seminar-stage the movement is from deep to hyper attention, with decreasing teacher activity [7].

The HY-DE philosophy and attention-training strategy have a place in the typology and strategic arsenal of computer problem-solving approaches. And, if it is there, we have the possibility to influence the shift between the forms of attention.

1.5 Concept-based Problem-Solving: a Dream in Computer Problem-Solving

In 1954 György Pólya published his well-known book entitled "How To Solve It. A New Aspect of Mathematical Method." [23], and has been proved effective and efficient in the meantime in various subjects and sciences. In his book, Pólya set up four steps of real-world problem-solving, which are summarized as follows:

- First. Understanding the problem.
- Second. Devising a plan. Find the connection between data and the unknown. You may be obliged to consider auxiliary problems if an immediate connection cannot be found. You should obtain eventually a plan of the solution.
- Third. Carrying out the plan.
- Fourth. Looking back. Examine the solution obtained. (pp. xvi-xvii)

According to Pólya's problem-solving approach, the first step is the understanding of the problem. In the present frame, this is the phase when the original real-world problem must be broken up, and a new, a more abstract structure must be built, which is relevant to the interface where the solution is meant to be carried out. This is the first, speaking session in the process of problem-solving, where we must take notes, draw figures, connections, introduce notations, using as far as possible the terminology of both the subject matter and the solution-interface.

In the second phase, the planning takes place. Primarily, we must consider what previously built schemata can be applied and what novel items are involved and must be built. After setting up the places and the propositions of old and new knowledge items, the execution order of the solution process must be decided. This is the design session of problem-solving in our model.

The third step is the actual solution when the plan is carried out. In end-user problem-solving, the content must be presented in a form of a document, and following that commands must be carried out according to the plan. This is the writing phase of our model: creating and formatting the content, adding visual data, creating formulas, etc.

The fourth step is discussing/debugging. In this phase strictly, but not exclusively, the result must be evaluated: we must check whether the solution is correct or not. Furthermore, we must take into consideration whether we can find other solutions, and finally, the possibility of abstraction. This is the second speaking phase, when all the results must be checked, contrasted to our expectations, more effective and generalized solutions must be looked for. This implies more than using a concept; it involves the ability to select an appropriate approach from understood alternatives [19] [26]. The generalization of end-user solutions plays a crucial role in the problem-solving process since these documents are meant to substitute hardcoded, paper-based, typed, or handwritten documents. Considering this purpose, the major characteristic of the digital documents is changeability, modifiability. This intention is well served with the definition of correctly edited and formatted spreadsheet [10]-[12] [28] [29] and text documents [18].

In general, the concept-based problem-solving approach is a slow thinking process that requires both hyper and deep attention modes. At this point, the question arises, how a slow thinking method can be integrated into a rapidly changing digital world. We argue in Chapter 1.6 that instead of the widely accepted, exclusively utilized hyper attention mode, the combination of hyper and deep attention modes must be introduced, where fast thinking can come into action with a firm knowledge background, built up in slow thinking processes.

1.6 Concept-based Problem-Solving: the Reality

Panko, considering spreadsheets, claimed that "thinking is bad" [12]. His paper reveals that working on spreadsheet problems, end-users apply slow thinking, in other words, they engage surface-approach methods, creating erroneous documents of various types [10]-[12] [22]. He claims in order to avoid errors, fast thinking (so the presence of schemata [26] are emphasized here) should be applied, which is one of the major characteristics of experts [6] [30] [31]. At this point, we beg to disagree with Panko's thoughts to some extent. As mentioned above, thinking slow is not bad, when it is applied in the right situation. In fact, thinking slowly, that is, working in deep attention phase is essential in the process of learning and creating. By carefully reading Panko's paper, it was revealed that he found that only Levels 1 and 2 – surface approach navigation methods – are applied in spreadsheet activities (Figure 1). These two approaches are

- the information based – practiced interface management and
- the trial-and-error wizard-based – interface browsing.

They require slow thinking, but are not problem-solving, consequently, do not provide reliable solutions and results. When comparing Panko's finding and the four steps of Pólya's problem-solving, we can conclude that only the third step is activated, the others are ignored, which methods lead to surface navigation in hyper attention mode.

Similar patterns can be recognized in text management, in the two major text-based, end-user-handled document types:

- text documents – prepared with word processors and
- presentations [18].

End-users author, modify, disseminate a huge number of erroneous documents, without recognizing their mistakes, errors. An analysis of the documents reveals that almost exclusively, activities at Levels 1 and 2 – surface approach navigation methods – are carried out, strictly in the hyper attention phase. Ben-Ari entitled this approach as bricolage [32], which expression clearly demonstrates end-users' incomprehensibility of the matter and their non-existing problem-solving strategies.

In general, it was found that end-user computing lost or never has found or never wanted to find the four steps of Pólya's problem-solving approach. Only, the third step exists with all its consequences:

- High demand on human and computer resources – authors' time spent on problems, the number of agents participating in solving a particular problem, reader's and audience's difficulty of understanding, not allowing space for hyper attention.
- Erroneous solutions and results causing financial losses [10]-[12] [22] [28] [29] not claiming to deep attention.
- End-users being ignorant of their ignorance – over-confidence and not rarely arrogance, proudly disseminating erroneous documents [33] –, hyper attention is used in inappropriate situations.
- The reproductivity of bricolage due to the lack of well-qualified teachers [34] and teaching materials [18] [32], due to the myths of the digital natives [25], unproved and disseminated statements, based on only hyper attention practice [34].

2 Presenting Errors

2.1 Utilizing Erroneous Documents

As detailed, erroneous documents have flooded both the private and the business spheres as a consequence of the shift from a textual universe to the digital world. To worsen the case, most of them are uploaded on the Internet, available for anyone, presenting samples and allowing their faster reproduction. The question is how we can stop and/or reverse this undesired progress. One solution is education, where these authentic sources can serve as practicing materials to demonstrate the errors and their consequences. The method is borrowed from other sciences and teaching programming languages. Especially, in the latter, erroneous programs are frequently used in introductory programming classes to lighten the burden of novices writing programs and to prepare them for debugging throughout their lives. Another possibility of calling attention to the problem is presenting at conferences that are open to such challenges.

The advantage of using authentic erroneous documents in education is, beyond providing inexhaustible sources, their effect on students. The force of authenticity plays a crucial role. First, the shock when the original authors are recognized, not rarely highly qualified persons, in publicly available materials and later on the proudness of "I-can-do-better". Finally, we cannot leave out the motivating aspect of the content. Any subject can be found which students are interested in, matches their age, gender, their background knowledge, and the aims of their classes.

In spite of the numerous advantages of using authentic erroneous documents for educational purposes, we can hardly find examples of good practices. As mentioned, in end-user computing and teaching end-user computing, almost exclusively low-mathability approaches are present, which focus on tools in hyper attention and slow thinking modes, without any thought of preparing and presenting real-world problems in classes. Consequently, the didactics of this approach is underrepresented either in scientific discussions or in the everyday classroom experience.

2.2 Collecting Erroneous Documents

2.2.1. Finding Keywords

One of the major characteristics of concept-based problem-solving and teaching is that authentic or adequate problems must be presented and solved. The major source of erroneous documents is the Internet, where millions of documents are available with the agreement of the authors by the act of uploading them.

The first concern of finding erroneous documents on the Internet is how to convince browsers to find and present only them. Teaching experience proves that quite often students and even parents find this task unsolvable, arguing that they typed in the search field "erroneous Word document" and its numerous variations, but no matches came along.

The problem with the "erroneous Word document" or "erroneous presentation" as keyword is that this search is not planned. Again, the first two steps of concept-based problem solving is ignored and end-users mindlessly jumped into the third step: "executing". In this case, the search is carried out in hyper attention mode. By analyzing the task – switching to deep attention mode –, it should become obvious that the keyword "erroneous" would not provide results. Authors do not upload erroneous documents intentionally for at least two reasons: (1) non-finished works are usually not uploaded, especially not with the title "erroneous", and (2) authors are not aware of their errors, consequently, they cannot express explicitly this feature of their documents.

2.2.2. Searching for the Suitable File Types

The second concern of this search is the type of document. Again, teaching experience proved, that searching for a specific type is not obvious and practiced, in spite of its frequency. In most of the browsers by default, the searching of the whole Internet is offered (Figure 2), regardless of file types, and only after launching the first search, special search is available. However, these offers are rather media-type and not file-type orientated. This setup of browsers leads to the misconception – built up again, on hyper attention activities and practice – that typing the extension into the search field would provide the desired document formats (Figure 3).

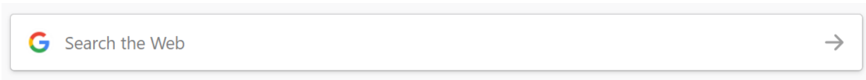


Figure 2
The default search option on launching a browser

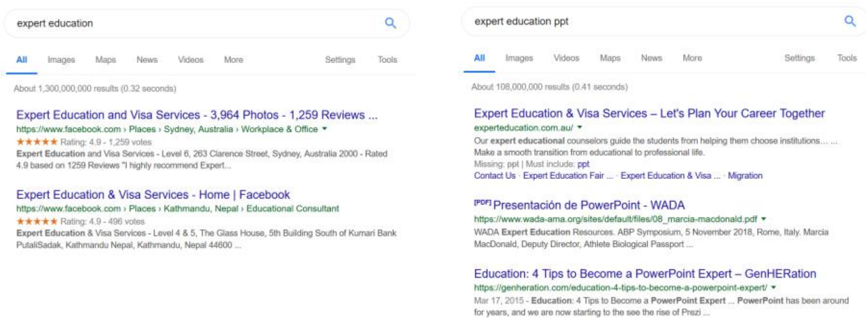


Figure 3
A preliminary search in the selected subject (left) and its extended search based on the misconception that special file types are available by typing the file extension into the search field (right)

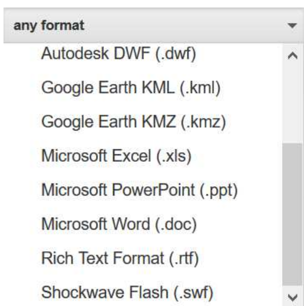


Figure 4
An extract of the file types offered by Google Advanced search

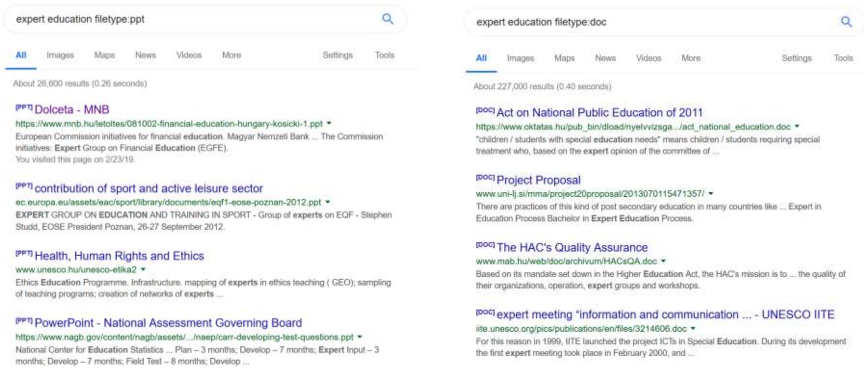


Figure 5

Searching for old PowerPoint and Word documents, with the extension of ppt and doc, respectively

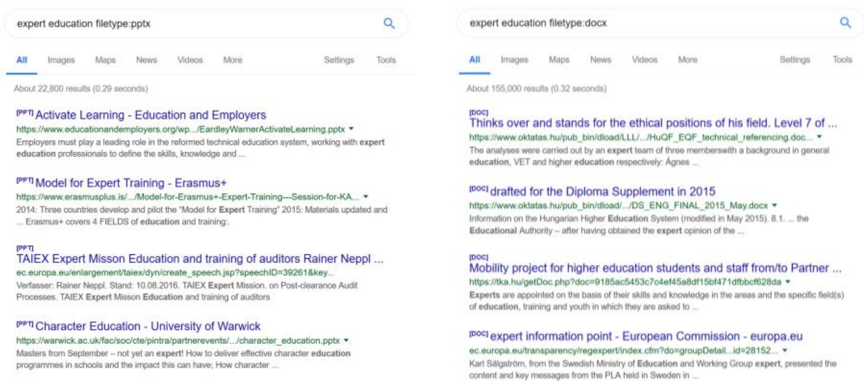


Figure 6

Searching for newer PowerPoint and Word documents, with the extension of pptx and docx, respectively

As mentioned, the content – the subject of the search – must be clarified in the first step of the problem-solving process (Figure 5, Figure 6), in the present case, the error types that we are looking for. Without conscious and meticulous design of the content, the search will lead to endless browsing without any result. Being aware of the teaching or research goals the task seems feasible.

After all these technical details and thousands of listed documents, comes the discussion phase of problem-solving, considering the content. The discussion requires both hyper and deep attention mode, where tens or hundreds of documents must be analyzed to find the document(s) which match our requirements the best. At this point of the discussion phase, we must realize that not finding the exact match is not because there are no errors in the documents, but the other way around, they contain so many errors that finding only the specified one is challenging (for details on error classification see [18]).

In education, the students' mental development, background knowledge, age, the field of interest, the aims of the classes, etc. must be taken into consideration when a document search is launched.

3 Designing Documents

One of the most demanding subjects of end-user computing is the handling of design-based styles (Figure 7, Figure 8).

3.1 Headings as Built-in Styles

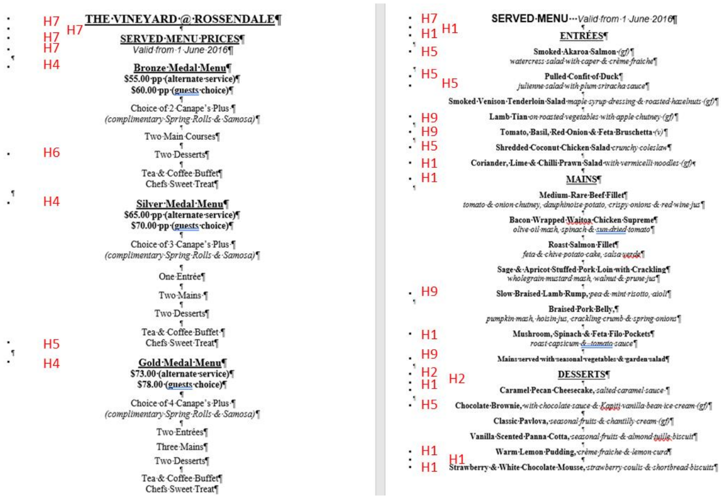


Figure 7

An incorrect style-combination is applied to a document. The red letter-number combinations indicate the header styles. The indented marks indicate those styles which are applied to empty paragraphs[2].

Software companies are rather known for selling their products by understating their complexity and advertising the user-friendly approach. Even though these programs cannot avoid offering design tools for helping uneducated users, the question is how end-users can apply these features, how they make the best of these offers. In this paper, not the different styles and their quality are our concern, but how end-users can apply effectively the built-in styles, nonetheless their own?

[2] na "THE VINEYARD @ ROSSENDALE". Retrieved February 24, 2019. from https://www.rossendale.co.nz/?wpdmact=process&did=MjUuaG90bGluaw==. A4 SERVED MENU. inc prices jun 2016.doc.

Microsoft Word offers several levels of headings guiding and inviting end-users for their use. “Headings make text stand out and help people scan your document. … The simplest way to add headings is with heading styles. Using heading styles means you can also quickly build a table of contents, reorganize your document, and reformat its design without having to manually change each heading's text.” [36].

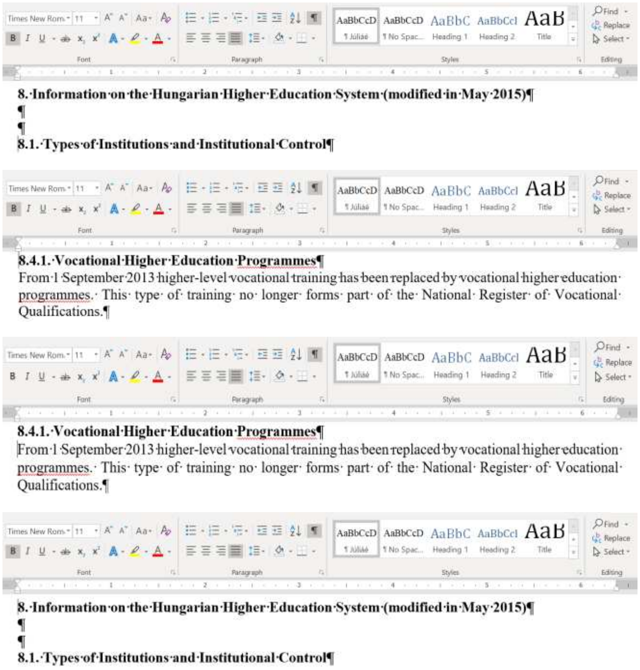


Figure 8

Incorrect use of costume style that does not allow the creation of an automated Table of Contents[3]

In Figure 7 a completely inconstant use of heading styles is presented. First, the styles are applied arbitrarily: there is no connection between the applied style and the content. Even empty paragraphs earned styles of different levels. On the other hand, the formats of the built-in styles are modified: the same levels got different formats (e.g. H7 style is applied with different formats for each instance). It is obvious that the author of the sample of Figure 7 did not read or did not understand the extract from “Add a heading” [36] but used the styles at the hyper-attention phase.On the contrary, Figure 8 is a sample of using only one custom style everywhere in the document, which is another form of hyper attention mode,

[3] na “Information on the Hungarian Higher Education System (modified in May 2015)”. Retrieved February 24, 2019 from https://www.oktatas.hu/pub_bin/dload/kepesitesek_elismertetese/DS_ENG_FINAL_2015_May.docx.

the lack of design, deep attention. It is obvious from the extract that this document is part of a more complex one, which requires a Table of Contents. With these formats creating a Table of Contents of high-operational-use is impossible. Furthermore, the document runs with manual numbering (Chapter 3.2), which further lowers the level of operational use [18]. Both examples demonstrate that without activating the deep-attention phase, only low-operational-use documents can be created, with all its consequences.

3.2 Manual Numbering

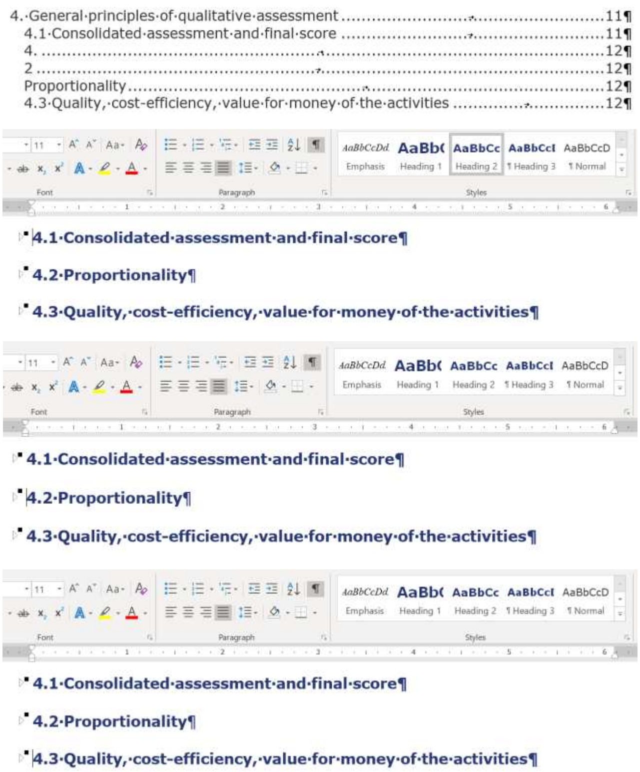


Figure 9

An incorrect use of built-in style which leads to incorrect automated Table of contents.[4]

Another form of distraction is, when the heading style is modified, and the Table of Contents cannot be created based on the assumed style (Figure 9). We also

4 na "Information on the Hungarian Higher Education System (modified in May 2015)". Retrieved February 24, 2019 from https://www.oktatas.hu/pub_bin/dload/kepesitesek_elismertetese/DS_ENG_FINAL_2015_May.docx.

must note that all the selected examples consist of manual numbering. As such, the numbering is not part of the formatted session and the style, consequently, is handled as any other character string of the text. Manual numbering, beyond not being part of the style, has the usual consequence of low-operational-use, meaning in this case that if any modification is made in the numbering, all the consecutive, connected sessions must be renumbered, which is a low-mathability approach to the problem.

3.3 Built-in Presentation Styles

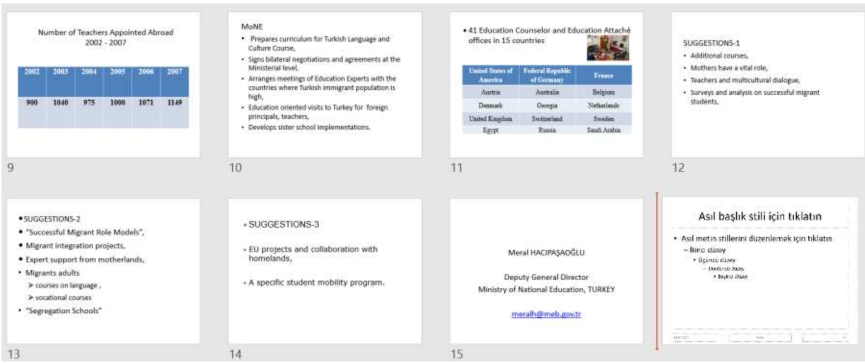


Figure 10

Various contents using the same Title and Content layout. The comparison of the slides with the Slide Master of Title and Content layout (lower right corner) clearly reveals that not only the selection of layouts is incorrect but also the use of the layout.[5]

Designing a presentation follows the same concept-based problem-solving method as detailed above. Analyzing presentations reveals that we are faced with the same problem: end-users only apply the third step of problem-solving, without giving a thought to design and discussion. All this happens, despite the effort of Microsoft. In PowerPoint, similar to Word, they set up layouts for various contents in the hope of convincing end-users to apply them. The documents found on the Internet clearly reveal that end-users, in general, are so hyper-attention-addicted that they do not even recognize when such obvious help is offered.

One of the most frequently met style errors in presentations is that regardless of the content the same layout is applied to the slides (Figure 10). Another form of distraction is when a selected style is misused. In the sample of Figure 10 the layout is completely ignored. In Figure 12 the background is set up in the Slide

---

[5] M. Haciapsaqlou "Turkish Migrants and Education" Ministry of National Education, TURKEY. Retrieved February 24, 2019 from http://www.oecd.org/education/innovation-education/41563670.ppt

Master and applied correctly, however the paragraph and character formats of the layout are ignored on the slides. All the slides are formatted individually, manually, not leaving space for the utilization of styles, this again is a low-mathability solution.

United States of America	Federal Republic of Germany	France
Austria	Australia	Belgium
Denmark	Georgia	Netherlands
United Kingdom	Switzerland	Sweden
Egypt	Russia	Saudi Arabia

Figure 11

Another form of ignoring style is presented in the table (Figure 10, Slide 11). Instead of using a list, a table style with a header row is applied, suggesting that the first row is the title. However, analyzing the content it is revealed that all the countries listed in the table play the same role, consequently, the selection of this style is not correct.

3.4 Considering the Target Group

One further aspect of planning presentations is the amount of content. This point of view plays a crucial role in any text-based document forms, however, presentations are the worst affected. It is a common practice that presentations hold the undesired amount of content, not considering the perceptive cognitive load of the audience.

To find a balance between the content and its presentation form requires a concept-based problem-solving approach by the author, which can only be carried out in deep-attention mode. On the other hand, the essence of the presentation is that it requires a hyper-attention mode from the audience. This paradox and the difficulty to resolve it can be an explanation to frequently encountered poorly designed presentations.

This paradox – creating in the deep attention phase by the author → perceiving in the hyper attention phase by the audience – weights a huge cognitive load on the authors, of which they usually are not aware of.

Figure 12

Slides with Title and Content layout seem to use the same formats. The comparison of the formats of the slides and the Slide Master of the Title and Content layout (lower right corner) reveals that the background is set up in the Slide Master, however, the paragraph and characters formats of the Slide Master are completely ignored and all the slides formatted individually.[6]

Conclusions

The present paper details how the different attention and thinking modes are connected to high- and low-mathability 'end-user computer problem-solving approaches'. Our investigation applies the HY-DE model framework, with its forms on the hyper and deep attention alterations and ratios. According to this model, the shift from hyper to deep attention requires necessary, well-designed teacher guidance through a mixed-phase of the two attention types.

– "I can do it, that's enough." – was our primary concern. However, the paper reveals that we must reformulate the slogan, and ask the question – How we can do it well. On the one hand, expressing that there is a difference between doing and doing well, on the other, we cannot do it alone, proper guidance from teachers are crucial for working effectively and efficiently.

The discussion is started with the presumed behavior of end-users, fulfilling the marketing slogans of software companies, and ends with the products of "real" end-users, wasting both human and computer resources, carrying out these outputs primarily guided by their ignorance. This ignorance is expressed in at least two forms: (1) the creation and modification of erroneous digital documents and (2) the dissemination of them. We searched for explanation and resolution to this paradox and found that the primary reason for the unexpected behavior of end-users is the selection of problem-solving methods. They prefer working with low-

[6] D. Kosicki "European Commission initiatives for financial education" Magyar Nemzeti Bank. Conference on Financial Awareness. 2 October 2008, Budapest. Retrieved February 24, 2019 from http://www.oecd.org/education/innovation-education/41563670.ppt.

mathability approaches, surfing the interfaces of the programs in hyper attention mode without considering the specialties and details of the problems. With this tool-centered approach, end-users do not give thoughts to analysis, design, and discussion, in short, for deep attention problem-solving methods. One further concern of the topic is the selection of thinking modes. In previous works, it was found that thinking might be the explanation for this immense number of erroneous documents. However, we argue that deep attention mode cannot be left out, but should be applied along with hyper attention mode by finding the right proportion and situation, leaving also space for high-mathability computer problem-solving.

References

[1] P. Baranyi, A. Csapo A., & G. Sallai: Cognitive Infocommunications (CogInfoCom), Springer International Publishing Switzerland, 2015, p. 191 (978-3-319- 19607-7, http://www.springer.com/us/book/ 9783319196077#aboutBook

[2] P. Baranyi & A. Csapo: Definition and Synergies of Cognitive Infocommunication, Acta Polytechnica Hungarica, 2012, Vol. 9, No. 1, pp. 67-83 (ISSN 1785-8860)

[3] P. Baranyi & A. Gilányi: Mathability: Emulating and Enhancing Human Mathematical Capabilities, The 4th IEEE International Conference on Cognitive Infocommunications, Budapest, Hungary, 2013, pp. 555-558, DOI=https://ieeexplore.ieee.org/document/6719309/

[4] TechTerms. End User, https://techterms.com/definition/enduser [13.13.2020.]

[5] Merriam-Webster. End user, https://www.merriam-webster.com/dictionary/end%20user [13.03.2020.]

[6] D. Kahneman: Thinking, Fast and Slow, New York, Farrar, Straus; Giroux, 2011

[7] E. Dani: The HY-DE Model: An Interdisciplinary Attempt to Deal with the Phenomenon of Hyperattention, Journal of Systemics, Cybernetics and Informatics, 2015, 13 (6), pp. 8-14

[8] K. Chmielewska and A. Gilanyi, “Educational context of mathability,” Acta Polytechnica Hungarica, Vol. 15, No. 5, pp. 223-237, 2018

[9] K. Chmielwska and A. Gilányi, “Computer assisted activating methods in education,” in 10th IEEE International Conference on Cognitive Infocommunications (CogInfoCom). IEEE, 2019

[10] R. R. Panko: What We Know About Spreadsheet Errors, Journal of End User Computing. Special issue on Scaling Up End User Development, 2008, (10) 2, pp. 15-21

[11] R. R. Panko & S. Aurigemma: Revising the Panko-Halverson taxonomy of spreadsheet errors, Decision Support Systems, 2010, (49) 2, pp. 235-244

[12] R. R. Panko: The Cognitive Science of Spreadsheet Errors: Why Thinking is Bad, Proceedings of the 46th Hawaii International Conference on System Sciences, Maui, Hawaii, 2013, pp. 1-10

[13] K. N. Hayles: Hyper and Deep Attention: The Generational Divide in Cognitive Modes [04.05.2018.]

[14] M. Csernoch & E. Dani: Data-structure validator: an application of the HY-DE model, The 8th IEEE International Conference on Cognitive Infocommunications, Debrecen, Hungary, 2017, pp. 197-202

[15] P. Biró & M. Csernoch: The mathability of computer problem solving approaches, The 6th IEEE International Conference on Cognitive Infocommunications, Győr, Hungary, 2015, pp. 111-114, DOI=http://doi.org/10.1109/CogInfoCom.2015.7390574

[16] P. Biró & M. Csernoch: The mathability of spreadsheet tools, The 6th IEEE International Conference on Cognitive Infocommunications, Győr, Hungary, 2015, pp. 105-110, DOI= http://doi.org/10.1109/CogInfoCom.2015.7390573

[17] M. Csernoch: Thinking Fast and Slow in Computer Problem Solving, Journal of Software Engineering and Applications, Vol. 10, No. 01 (2017), Article ID:73749, pp. 30 10.4236/jsea.2017.101002

[18] M. Csernoch: Do You Speak and Write in Informatics? 10th International Multi-Conference on Complexity, Informatics and Cybernetics (IMCIC 2019), Orlando, Florida, USA 12-15 March 2019, pp. 157-152

[19] IEEE&ACM Report. Computer Science Curricula 2013. The Joint Task Force on Computing Curricula Association for Computing Machinery (ACM) IEEE Computer

[20] J. B. Biggs & K. E. Collis: Evaluating the Quality of Learning: The SOLO Taxonomy, 1982, Academic Press, New York

[21] J. M. Wing: Computational thinking, Communications of the ACM, 49(3) 2006, 33, DOI= http://doi.org/10.1145/1118178.1118215

[22] EuSpRIG, European Spreadsheet Risk Interest Group (2020), EuSpRIG Horror Stories. Retrieved 28 February 2020 from http://www.eusprig.org/horror-stories.htm

[23] Gy. Pólya: How To Solve It. A New Aspect of Mathematical Method. Second edition, Princeton University Press, Princeton, New Jersey, 1945

[24] M. Prensky: Digital Natives, Digital Immigrants, From On the Horizon, MCB University Press, Vol. 9 No. 5, October 2001, http://www.marcprensky.com/writing/Prensky%20-

%20Digital%20Natives,%20Digital%20Immigrants%20-%20Part1.pdf [03.03.2018.]

[25] P. A. Kirschner, & P. De Bruyckere: The myths of the digital native and the multitasker, Teaching and Teacher Education, 2017, 67, pp. 135-142

[26] J. Sweller, P. Ayres, & S. Kalyuga: Cognitive Load Theory. Berlin: Springer, 2011

[27] M. Csernoch & P. Biró: Edu-Edition Spreadsheet Competency Framework, Proceedings of the EuSpRIG 2017 Conference "Spreadsheet Risk Management", Imperial College, London, 2017, pp. 121-136

[28] P. L. Bewig: How do you know your spreadsheet is right? Principles, Techniques and Practice of Spreadsheet Style, 2005. https://arxiv.org/ftp/arxiv/papers/1301/1301.5878.pdf [12.11.2019.]

[29] Z. Przasnyski, L. Leon, & K. C. Seal: In Search of a Taxonomy for Classifying Qualitative Spreadsheet Errors, Proceedings of EuSpRIG 2011 Conference, “Spreadsheet Governance – Policy and Practice”, 2011

[30] National Research Council How People Learn: Brain, Mind, Experience, and School: Expanded Edition, National Academy Press, Washington, D.C., 2000

[31] C. Roda: Human Attention in Digital Environments, Cambridge University Press, 2011

[32] M. Ben-Ari: Bricolage Forever! PPIG 1999, 11th Annual Workshop, Computer-Based Learning Unit, University of Leeds, UK, 1999 http://www.ppig.org/papers/11th-benari.pdf [12.03.2015.]

[33] J. Kruger, & D. Dunning: Unskilled and Unaware of It: How Difficulties in Recognizing One's Own Incompetence Lead to Inflated Self-Assessments, Journal of Personality and Social Psychology, 1999, 77 (6), pp. 1121-34

[34] R. Lister: After the Gold Rush: Toward Sustainable Scholarship in Computing, Proc. Tenth Australasian Computing Education Conference (ACE2008), Wollongong, Australia, CRPIT Volume 78 – Computing Education 2008

[35] N. Garrett: Textbooks for Responsible Data Analysis in Excel, Journal of Education for Business, 0: 1-6, 2015, Taylor and Francis Group, LLC, DOI: 10.1080/08832323.2015.1007908

[36] Microsoft, Add a heading, https://support.office.com/en-us/article/add-a-heading-3eb8b917-56dc-4a17-891a-a026b2c790f2 [02.04.2019.]

Cross-Curricular Connections and Knowledge-Transfer Elements in Data Management

Tímea Katalin Nagy[1], Mária Csernoch[1], János Máth[2]

[1]Faculty of Informatics, University of Debrecen, Kassai út 26, 4028 Debrecen, Hungary; nagy.timea@inf.unideb.hu; csernoch.maria@inf.unideb.hu

[2]Faculty of Arts and Humanities, University of Debrecen, Komlóssy út 45-47, 4032 Debrecen, Hungary; math.janos@arts.unideb.hu

Abstract: In this study, we present the items and the results of a complex task in which the data management knowledge of students was measured in the 2018 academic year. Our results obtained from statistical analyses are in line with previous studies that analyzed informatics/computer science coursebooks and concluded that current books favour surface-based approaches and consequently, are unsuitable for developing students' computional thinking skills. Furthermore, our study found evidence that contrary to Prensky's unproven claim, digital children also need teachers' support in spreadsheet environments to learn effective computer problem-solving strategies, avoid common spreadsheet errors, and acquire basic algorithmic and programming skills. Our results, clearly show that the widely accepted low-mathability approaches should be excluded from the teaching-learning process due to their ineffectiveness, and should be replaced by high-mathability approaches, which support knowledge-transfer, effective end-user computing, and introductory studies to "serious" informatics/computer sciences.

Keywords: Informatics; algorithmic skills; computational thinking; computer problem-solving; knowledge-transfer

1 Introduction

Informatics as a school subject is compulsory in grades 6-10, with one class a week. Additional lessons and courses in lower grades are optionally available. Previous measurements, classroom experiences, and teacher interviews have shown that the current National Base Curriculum and the Hungarian frame curricula in informatics contain several vague, difficult-to-understand terms that make teachers' planning difficult. We also found that informatics/computer sciences (ICT) education is surface-based since most ICT teachers focus on teaching interfaces instead of problem-solving [1] [2] [3]. Within the frame of this decontextualized [30] low-mathability approach [1], students carry out erroneous slow thinking [21] activities instead of highmathability [1] concept-based real-world problem-solving [3] with the right proporsion of fast and slow thinking [21] [28].

Computational thinking is mentioned as the fourth basic skill in the current Hungarian basic educational documents [4]-[15] along with Reading wRiting, and aRithmatic [16]. However, it is not clear from these documents how ICT education supports the development of an algorithmic approach, or cross-curricular and intra-ICT knowledge transfer [17].

In addition to subjective classroom experiences [18] and tool-centered low-mathability tests [28], we needed an objective high-mathability, contextualized measurement tool to ascertain or disprove students' behaviors and the presence or absence of knowledge elements associated with them, during classes and computer problem solving [18].

For this reason, we designed a mini-competence test to identify students' knowledge transfer elements and to measure students' computational thinking and algorithmic skills in traditional and non-traditional programming environments. The purpose of the assignments was to directly measure how students apply the knowledge acquired in other subjects and in other areas of informatics in new environments and how they recognize algorithms and common knowledge transfer elements.

2 Methodology

The essence of our mini-competence test is to measure the students' level of computational thinking skills which is in complete accordance with our high-mathability teaching-learning approach [44]. According to our knowledge transfer based method, the primitive data types are introduced in spreadsheet management accompanied with real world tables, and then this knowledge can be transferred to database management and "serious" programming.

3 Sample

Elementary and secondary students from the 7-10th grades completed the test all over Hungary. A total number of 8880 students from 93 schools returned the completed tests [19] [20]. A further 269 tests were returned from the 5-6th and the 11-12th grades (additional tests) (Table 1).

Table 1
The sample of the mini-competence test, measuring the students' knowledge-transfer skills

Grade	Sample
7	1561
8	1639
9	3059
10	2352
additional tests	269
total	8880

4 Hypotheses

In this study, we analyze students' basic knowledge transfer skills in data management and seek answers as to how different ICT topics prepare for data management studies. Our primary goal is to demonstrate how knowledge transfer tasks support schema construction through reliable, fast-thinking [21][21]. Taking all these aims into consideration, we have set up the following hypotheses related to the subject of data management:

[H1] In cases in which students recognize the integer correctly they recognize the real data type also correctly in a significantly greater extent.

[H2] In cases in which students recognize the largest number in a vector, they also recognize the smallest number also correctly in a significantly greater extent.

[H3] In cases in which students recognize both the integer and the real datatypes, they can also recognize the smallest and largest numbers in the vector also correctly to a significantly greater extent.

[H4] The students' results significantly improve by the end of 10^{th} grade.

The Description of The Task

Based on the current Hungarian ICT frame curricula, students can interpret data, and recognize different elementary and complex datatypes [5] [6] [7]. In accordance with the frame curricula, the DigComp 2.1 at Foundation Level 2 requires the same skills and emphasis as at Levels 3 and 4 (Figure 1).

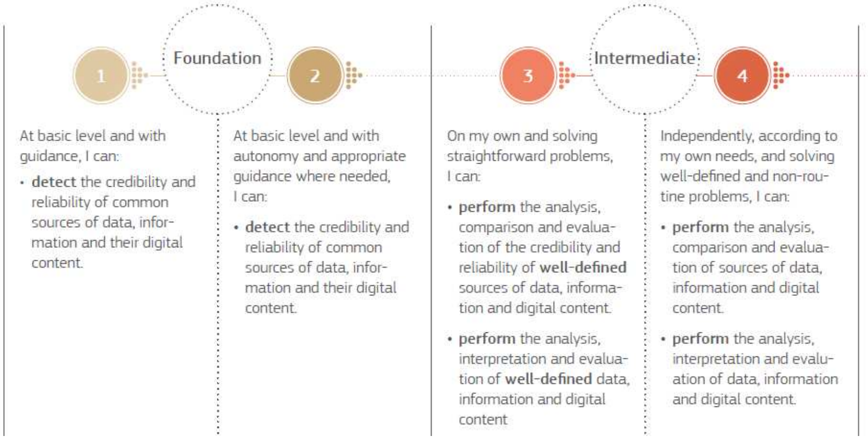


Figure 1

DigComp 2.1, Competence area 1: Information and data literacy 1.2 Evaluating data, information, and digital content. To analyse, compare and critically evaluate the credibility and reliability of sources of data, information and digital content. To analyse, interpret and critically evaluate the data, information; and digital content. Proficiency Levels 1-4.

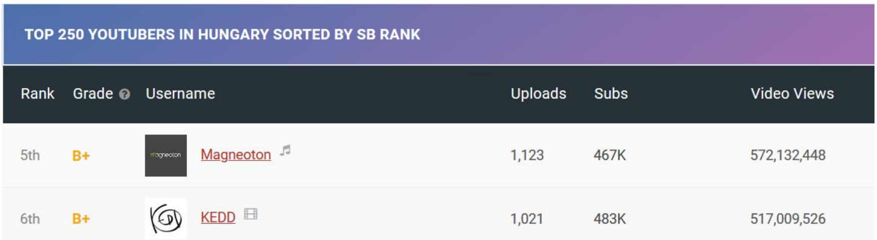
To measure the above-mentioned basic data management skills, our test consists of a task considering the datatypes, the number of data records in the table, and the minimum and maximum values.

	A	B	C	D
1	Username	Uploads	Subs	Views
2	VamosART	484	1,107,555	226,195,766
3	Videómánia	338	833,23	254,545,702
4	PamKutya	120	809,866	223,441,355
5	LetsGoMartin	176	725,638	162,798,559
6	TheVR	1,062	592,675	213,550,948
7	luckeY	1,183	561,13	150,341,428
8	Peter Gergely	100	548,241	79,713,757
9	Scribble Netty	159	546,049	74,234,471
248	Szilvaglam	87	61,899	3,918,538
249	rance flow	524	61,863	53,275,385
250	KIS GRÓFO (official)	9	61,65	30,712,031
251	KODIAK	736	61,467	14,599,194

Figure 2

DigComp-compatible task for detecting the credibility of sources of data, information, and their digitial content, performing the analysis, interpreting, and evaluating well-defined data, information, and digital content

To complete the items of the task, an adapted spreadsheet table was provided, converted from the "Top 250 YouTubers Channels in Hungary" webtable [24] [24] [22]. In the teaching-learning process, the use of authentic and adapted data sources plays a crucial role since tables with real contents have a motivating effect on students [25] [25]. On the other hand, it has also been shown that the teaching of informatics and spreadsheets is rather interface and tool-oriented, and applies decontextualized data [26] [28] [29] [30]. Measuring the effectiveness of these contradictory approaches and proving how they support the development of students' computer problem-solving strategies and computational thinking skills are the aims of the present paper.


TOP 250 YOUTUBERS IN HUNGARY SORTED BY SB RANK

Rank	Grade	Username	Uploads	Subs	Video Views
5th	B+	Magneoton	1,123	467K	572,132,448
6th	B+	KEDD	1,021	483K	517,009,526

Figure 3
Top 250 YouTubers Channels in Hungary

The complexity of this task derives mainly from the semantics of the comma; whereas in English the comma functions as a thousand-separator character, in Hungarian the comma stands for the decimal character. Although the country on the SocialBlade website can be selected, the syntax of the content does not follow this selection, nor does it match the rules of the selected language. The Uploads column contains the number of uploads and the Video Views the number of views, which are integers, considering the semantics of these data.

Task-Solving with Datatypes

In Excel and most modern spreadsheet environments, datatypes can easily be recognized based on the default alignment settings. According to these arrangements, the following default alignments are widely accepted (Table 2).

Table 2
The default datatype-alignments in spreadsheet programs

datatype	alignment
text	left
integer	right
real number	right
logical	center
errors	center
datatype	alignment

The appearance of graphical interfaces has undoubtedly simplified the use of computers for those who do not have advanced IT skills. However, the interpretation of these graphic “messages“ requires competencies that can be significantly accelerated in well-organized and planned educational environments. A further aim of our testing was to investigate whether it is necessary to teach data management in spreadsheet environments [30] [32] [22] or it is sufficient to allow students to explore and operate alone in the graphic interfaces, bringing their native knowledge with them [33] [34].

	Integer	Real	Text	Logical	Date	None
Column A	☐	☐	☒	☐	☐	☐
Column B	☒	☒	☐	☐	☐	☐
Column C	☐	☒	☒	☐	☐	☐
Column D	☐	☐	☒	☐	☐	☐

Figure 4

The first item of the task: selecting the datatypes of each column of the presented the table

Based on the adapted table (Figure 3), the datatypes of Columns A-D are presented in Figure 4. Column A is the simplest since it consists of the name of the youtubers; consequently, the datatype is text in this column. Considering the semantics of Columns C-D, the datatypes should be integers. However, one comma turns these integers into real numbers, while two commas are into strings. Consequently, Column B has both integers and real numbers, Column C a string and real numbers, and Column D strings, exclusively, due to the two commas.

The largest number in Column B is: ....

The smallest number in Column B is: ....

Figure 5

The items regarding the recognition of the minimum and the maximum values in column B

There are two other items related to the recognition of datatypes and contents, asking the minimum and maximum values in column B (Figure 5). The purpose of this open-ended task was to filter out random answers in the previous multiple-choice item.

The correct solution to the problem is that the values shown in column B have a maximum of 736 and a minimum of 1,062 in Hungarian and European Excel (Figure 6) (1.062 in English Excel).

	A	B	C	D
1	Username	Uploads	Subs	Views
2	VamosART	484	1,107,555	226,195,766
3	Videómánia	338	833,23	254,545,702
4	PamKutya	120	809,866	223,441,355
5	LetsGoMartin	176	725,638	162,798,559
6	TheVR	1,062	592,675	213,550,948
7	luckeY	1,183	561,13	150,341,428
8	Peter Gergely	100	548,241	79,713,757
9	Scribble Netty	159	546,049	74,234,471
248	Szilvaglam	87	61,899	3,918,538
249	rance flow	524	61,863	53,275,385
250	KIS GRÓFO (official)	9	61,65	30,712,031
251	KODIAK	736	61,467	14,599,194

Figure 6

The lowest (red) and highest (green) values in column B

The next question in the test focuses on the identification of the data records in the table. The table has 251 rows, of which the first row contains the field names, so the number of data records is 250, of which Rows 10-247 are hidden (Figure 7).

How many data rows (records) are in the table?...

Figure 7

To identify the data records in the table

Datatypes as the Knowledge-Transfer Elements of the Task

One of the most important cross-curricular knowledge-transfer elements of the table is the semantics of the comma, a problem that arises in the conversion process from webtable to datatable [26] [24] [35] [22]. The concept of datatype can also be considered an intra-informatics knowledge-transfer element because if students learn datatypes in a spreadsheet environment, this knowledge is transferable to further studies in database management and programming since in spreadsheet environments, due to their informative graphical interfaces, the subject of datatypes can be introduced effectively [26] [36] [37] [38]. It has also been proved that the combination of the functional language of spreadsheet programs and high-mathability approaches to spreadsheeting would serve as an introduction to programming [39] [40] [41]. Programming in spreadsheet environments combines the benefits of both the functional languages and the graphical interfaces, providing a programming tool for non-professional IT end-users.

A further cross-curricular relationship to mathematics is recognizable, as the set of natural numbers are extended to integers in elementary school in the 5^{th} grade, to decimal fractions in the $6\text{-}7^{th}$ grades, and to irrational numbers in the $7\text{-}8^{th}$ grades, thus introducing the concept of real numbers [42] [43].

5 Results

The number of answers considering the datatypes in Columns A-D is presented in Table 2. It was found that 65% of the students marked the datatype correctly in column A (5774 students). 3722 and 3374 students answered correctly that there are integers and real numbers in column B, respectively. In column C, 3711 and 1201 students marked correctly the real number and string datatypes. In column D, 1159 students answered correctly that the datatype is a string (Table 3).

Table 3

The number of responses to the datatypes in each column

	Integer	Real	Text	Logical	Date	None
Column A	642	485	5774	284	138	372
Column B	3722	3374	860	437	206	259
Column C	2138	3711	1201	793	311	392
Column D	2370	2596	1159	1078	290	650

In the minimum-maximum item, several students wrote 1183 for the largest and 9 for the smallest number (Table 5). Considering the incorrect answers, we can conclude that these students were unaware of the semantics of the comma; they recognized the comma as the thousand-separator character and did not realize the change in its role. The minimum value in column B was correctly answered by 865 students, which is 10% of the sample, and the maximum by 963 students, 11% of the sample.

Considering the number of data records, 17% of the students answered incorrectly that there are 251 data records, while 357 answered 12, 365 answered 13, and 954 answered 3 or 4 (Table 3).

The correct number of data records was answered by 822 students, which is only 9% of the sample (Table 4). We also must call attention to the high number of students who skipped this task.

Table 4
The number of responses to the data records

	Number of records					
Answer	250	251	12	13	3 or 4	ignored
Marked	822	1500	357	675	959	4567
%	9	17	4	8	11	51

Those students who answered 251 realized that there are hidden rows in the table, but did not separate the row of field names from the data records. Those who answered 12 are aware of the concept of data record and field name but did not notice that there are hidden rows in the table. In contrast, those who responded 13, neither noticed the hidden rows nor the row of the field name. All they did was count the number of rows in the sample table presented. Those who answered 3 or 4, did not know the definition of either a data record or a field name.

Table 5
The number of responses to the lowest and highest values

	Maximum			Minimum		
Answer	736	1183	ignored	1,062	9	ignored
Marked	963	4908	3009	865	5010	3005
%	11	55	34	10	56	34

As presented in Figure 8, there are 933 students who recognized both the integer and the real number datatypes in column B, while 815 students found the minimum and maximum values correctly. The comparison of the pairs revealed that there are only 154 students who recognized both the datatypes and the minimum and maximum values. This result shows that only 154 students were able to combine all the knowledge elements required to complete the task – integer and real datatypes, the syntax and sense of the comma in numbers in Hungarian –, which is 1.7% of the sample.

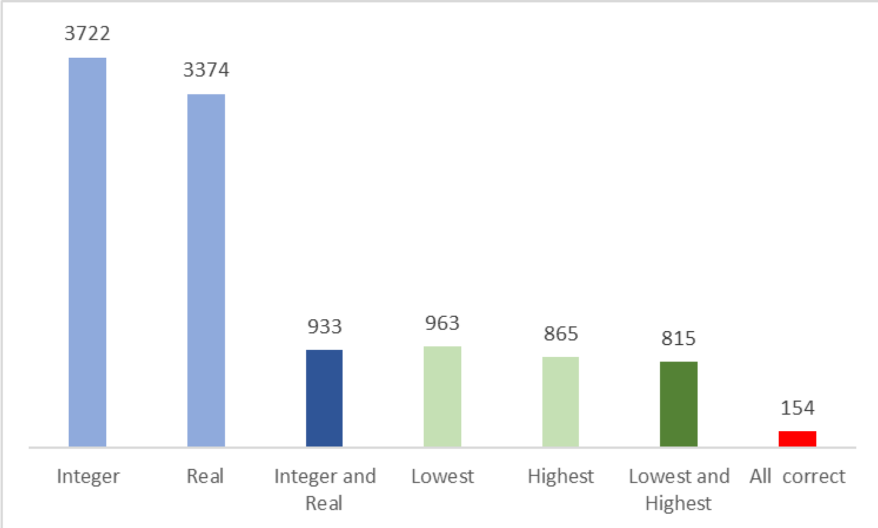


Figure 8
The number of students who correctly recognize the datatypes, the extrema in the datatable

We performed an independence and homogeneity test to reveal the relationship between the recognition of integer and real numbers, and found that they make errors symmetrically, which means that if they recognize one datatype, there is no guarantee that they also recognize the other (Table 6). Based on this outcome, we must reject hypothesis H1.

In the comparison of the results of the different age groups, we found that there is only a minimal improvement by the end of grade 10 ($p < 0.001$). This result failed to validate hypothesis H4.

Table 6
Testing independence in SPSS: those who marked the integer also marked the real datatype

		B_real		Total
		not marked	marked	
B_real	not marked	2714	2441	5155
	marked	2788	3374	3721
Total		5502	3374	8876

To confirm hypothesis H2, in cases in which students recognize the smallest value in Column B, they also recognize the largest. We made an independence test with the Chi-square test. Based on the result of the cross-table, we can say with a probability of 0.85 that hypothesis H2, due to the asymmetry of the matrix, is proved ($p < 0.001$) (Table 7).

Table 7

Chi-square test in SPSS: those students who found the smallest value could find the largest value, too

		minimum	
		ignored	marked
maximum	ignored	99.4%	0.6%
	marked	15.4%	84.6%

We can conclude that students who recognized one of the datatypes did not necessarily recognize the other. On the other hand, the analysis proved that if one of the extrema is found, so is the other. Based on these findings, hypothesis H1 is rejected, while H2 is proved. However, at this phase of the evaluation, it was not clear what the connection is between the datatypes and the extrema.

In hypothesis H3, we assumed that if the students recognize both datatypes, they are also able to tell the minimum and maximum values in column B. To prove or reject this hypothesis, we performed a Chi-square test. We found that a greater proportion of students recognized the extrema in cases in which they recognized the datatypes; however, no relation was found between the two variables ($p<0.001$) (Figure 9).

Figure 9 presents the results of the extremum-item in connection with the students' recognition of datatypes. Students not recognizing any of the datatypes have the lowest results in finding the extrema. Of those who recognized one of the datatypes, 10.6% recognized the minimum and 11.9% the maximum. Of those who recognized both datatypes, reached 18.1% and 19.3% in the minimum- and maximum-item, respectively.

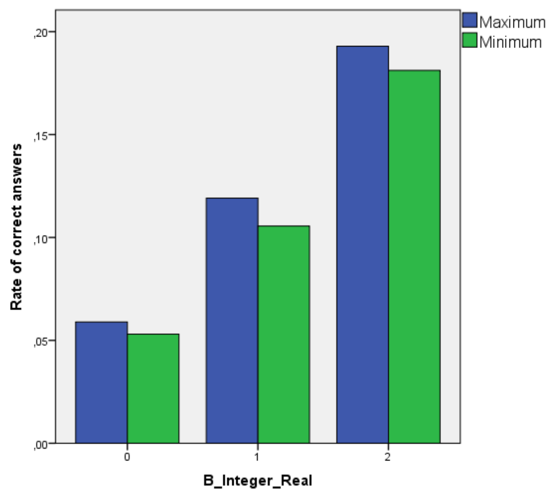


Figure 99

The percentage of students recognizing 0, 1, and 2 datatypes and solving the maximum (blue) and minimum (green) items correctly

6 Summary

Based on our testing procedure, we were able to prove that students make errors symmetrically in recognizing datatypes. This means that even though they recognize one datatype, the other will not necessarily be recognized. With this result, hypothesis H1 is rejected, so the recognition of one datatype does not imply the recognition of another.

An independence test was performed with the Chi-square test, which shows that, with a probability of 0.85, if students recognize the largest number in a vector, they also recognize the smallest number ($p<0.001$), thus confirming our H2 hypothesis.

Only 154 of the 8880 students can recognize both the datatypes and the minimum and maximum values, representing only 1.7% of the sample. We performed a Chi-square test to determine whether there was a relationship between the student's recognition of datatypes and extrema. It is not possible to draw a clear conclusion from the result, due to the low number of answers, but the tendency shows that after recognizing the datatypes, the students could determine correctly the lowest and highest values. However, we cannot prove our hypothesis with this test, so further measurements are needed for this purpose.

Conclusions

In this study, we presented the items of a task and their solutions, from a test focusing on the knowledge-transfer skills of 7-10th grade Hungarian students.

One aspect of the data management knowledge of students was tested in a spreadsheet environment since previous studies proved that high-mathability spreadsheet approaches would serve to teaching introductory data management and programming effectively and efficiently [39] [39] [44].

The results of our test, however, proved that low-mathability approaches to and self-taught experiences of spreadsheeting do not help students' understanding of datatypes and the problems connected to them [45]. These findings are in complete accordance with previous studies which claimed that decontextualized, interface- and tool-centered approaches do not help develop computational thinking skills, and consequently, effective problem-solving abilities [26] [36] [37] [38].

Our results, based on the students' performance in this knowledge-transfer focused test, might call attention to the lack of students' knowledge in end-user computing, especially in data management. Furthermore, it is also clearly shown that the widely accepted low-mathability approaches should be excluded from the teaching-learning process, and should be replaced by high-mathability approaches, which support knowledge-transfer, effective end-user computing, and introductory studies to "serious" informatics/computer sciences [46] [47].

Acknowledgement

This work was supported by the construction EFOP-3.6.3-VEKOP-16-2017-00002. The project was supported by the European Union, co-financed by the European Social Fund.

References

[1] Baranyi, P. & Gilanyi, A. (2013) Mathability: emulating and enhancing human mathematical capabilities. 2013 IEEE 4th International Conference on Cognitive Infocommunications (CogInfoCom), 2013, pp. 555-558, doi: 10.1109/CogInfoCom.2013.6719309

[2] Freiermuth, K., Hromkovic, J., & Steffen, B. (2008) Creating and Testing Textbooks for Secondary Schools. In Proceedings of the 3rd International Conference on Informatics in Secondary Schools - Evolution and Perspectives: Informatics Education - Supporting Computational Thinking, pp. 216-228, Berlin, Heidelberg: Springer-Verlag. DOI= http://doi.org/10.1007/978-3-540-69924-8_20

[3] Pólya, G. (1954) How To Solve It. A New Aspect of Mathematical Method. Second edition (1957) Princeton University Press, Princeton, New Jersey. Magyarul: A gondolkodás iskolája. Gondolat Kiadó

[4] NAT 2012 (2012) National Base Curriculum. In Hungarian: 110/2012. (VI. 4.) Korm. rendelete a Nemzeti alaptanterv kiadásáról, bevezetéséről és alkalmazásáról. Retrieved 07. 02. 2018. from http://ofi.hu/sites/default/files/attachments/mk_nat_20121.pdf

[5] OFI (2013a) Frame Curricula for primarry school grades 1-4. Optional subjects 1.3.3. In Hungarian: Kerettanterv az általános iskola 1-4. évfolyamára. Szabadon választható tantárgyak 1.3.3. Retrieved 07. 02. 2018. from http://kerettanterv.ofi.hu/01_melleklet_1-4/1.3.3_informat_1-4.doc

[6] OFI (2013b) Frame Curricula for primary school grades 5-8 Compulsory subjects 2.2.15. In Hungarian: Kerettanterv az általános iskola 5-8. évfolyamára. Kötelező tantárgyak 2.2.15. Retrieved 07. 02. 2018. from http://kerettanterv.ofi.hu/02_melleklet_5-8/2.2.15_informat_5-8.doc

[7] OFI (2013c) Frame Curricula for primary school grades 9-12. Increased number of classes 2.3.2. In Hungarian: Kerettanterv az általános iskola 5-8. évfolyamára. Emelt óraszámú kerettantervek 2.3.2. Retrieved 07. 02. 2018. from http://kerettanterv.ofi.hu/02_melleklet_5-8/2.3.2_informat_5-8.doc

[8] OFI (2013d) Frame Curricula for grammar school grades 9-12. Compulsory subjects 3.2.16. In Hungarian: Kerettanterv a gimnáziumok 9-12. évfolyama számára. Kötelező tantárgyak 3.2.16. Retrieved 07. 02. 2018. from http://kerettanterv.ofi.hu/03_melleklet_9-12/3.2.16_informat_9-12.doc

[9] OFI (2013e) Frame Curricula for grammar school grades 9-12. Increased number of lessons 3.3.6. In Hungarian: Kerettanterv a gimnáziumok 9-12. évfolyama számára. Emelt óraszámú kerettantervek 3.3.6. Retrieved 07. 02. 2018. from http://kerettanterv.ofi.hu/03_melleklet_9-12/3.3.6_informat_emelt_9-12_u.docx

[10] OFI (2013f) Frame Curricula for vocational school grades 9-12. Compulsary subjects 6.2.11. In Hungarian: Kerettanterv a szakközépiskolák 9-12. évfolyama számára. Kötelező tantárgyak 6.2.11. Retrieved 07. 02. 2018. from http://kerettanterv.ofi.hu/06_melleklet_9-12_szki/6.2.11_informat_9-10_sz.doc

[11] OFI (2013g) Frame Curricula for vocational school grades 9-12. Increased subjects 6.3.7. In Hungarian: Kerettanterv a szakközépiskolák 9-12. évfolyama számára. Emelt óraszámú kerettantervek 6.3.7 http://kerettanterv.ofi.hu/06_melleklet_9-12_szki/6.3.7_informat_emelt_9-12_sz_u.docx

[12] OFI (2013h) Frame Curricula for vocational school grades 9-12. In Hungarian: Kerettanterv a szakgimnáziumok 9-12. évfolyama számára. Retrieved 07. 02. 2018. from http://kerettanterv.ofi.hu/20160825_szakgimnazium.doc

[13] OFI (2013i) Frame Curricula for primary school grades 1-4. In Hungarian: Kerettanterv az általános iskola 1-4. évfolyamára. Retrieved 07. 02. 2018. from http://kerettanterv.ofi.hu/01_melleklet_1-4/index_alt_isk_also.html

[14] OFI (2013j) Frame Curricula for primray school grades 5-8. In Hungarian: Kerettanterv az általános iskola 5-8. évfolyamára. Retrieved 07. 02. 2018. from http://kerettanterv.ofi.hu/02_melleklet_5-8/index_alt_isk_felso.html

[15] OFI (2013k) Frame Curricula for grammar school grades 9-12. In Hungarian: Kerettanterv a gimnáziumok 9-12. évfolyama számára. Retrieved 07. 02. 2018. from http://kerettanterv.ofi.hu/03_melleklet_9-12/index_4_gimn.html

[16] Hildreth, G. (1947) Cumberland education series. Learning the three R's (2nd ed.) (P. L. Boynton, Ed.). Educational Publishers, Inc. https://doi.org/10.1037/14928-000

[17] Wing, J. M. (2006) Computational thinking. Communications of the ACM, 49(3), pp. 33-35

[18] Lister, R. (2008) After the Gold Rush: Toward Sustainable Scholarship in Computing, Proc. Tenth Australasian Computing Education Conference (ACE2008), Wollongong, Australia, CRPIT Volume 78 - Computing Education 2008

[19] Nagy T. (2018) Analyzing the Informatics Frame Curricula. In Hungarian: Az informatika kerettanterv elemzése. 3th International Interdisciplinary Conference 2018. Retrieved 28. 07. 2020. from

https://detep.unideb.hu/sites/default/files/upload_documents/kotet_interdisz_3.pdf

[20] Nagy T. & Csernoch M. (2018) Measuring the Level of Computer Problem Solving Abilities in Teaching Informatics. In Hungarian: Számítógépes problémamegoldás mérése az informatika órán. InfoÉra 2018, Zamárdi 2018. november 22-24., Retrieved 28. 07. 2020. from https://people.inf.elte.hu/szlavi/InfoDidact18/Infodidact2018.pdf

[21] Kahneman, D. (2011) Thinking, Fast and Slow. New York: Farrar, Straus and Giroux

[22] Csernoch, M. (2019) From webtables to datatables. EuSpRiG 2019 Conference, London. Retrieved 06. 08. 2020 from https://arxiv.org/pdf/2006.14694.pdf

[23] EuSpRIG, European Spreadsheet Risk Interest Group (2019) EuSpRIG Horror Stories. Retrieved 28. 07. 2020. from http://www.eusprig.org/horror-stories.htm

[24] Csernoch, M. & Dani, E. (2017) Data-structure validator: an application of the HY-DE model. 2017 8th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), 2017, pp. 197-202, doi: 10.1109/CogInfoCom.2017.8268242

[25] Gross, D., Akaiwa, F., & Nordquist, K. (2013) Succeeding in Business with Microsoft Excel 2013: A Problem-Solving Approach. Cengage Learning, Delhi

[26] Csernoch, M. (2014) Programming with Spreadsheet Functions: Sprego, In Hungarian, Programozás táblázatkezelő függvényekkel – Sprego, Műszaki Könyvkiadó, Budapest

[27] Garrett, N. (2015) Textbooks for Responsible Data Analysis in Excel. Journal of Education for Business, 0: 1–6, 2015, Taylor and Francis Group, LLC, ISSN: 0883-2323 print / 1940-3356 online, DOI: 10.1080/08832323.2015.1007908

[28] Csernoch, M. (2017) Thinking Fast and Slow in Computer Problem Solving. Journal of Software Engineering and Applications. Vol.10 No.01 (2017), Article ID: 73749, 30 pages 10.4236/jsea.2017.101002 pp. 1-31

[29] Csernoch, M., Biró, P. (2018) Edu-Edition Spreadsheet Competency Framework. Proceedings of the EuSpRIG 2017 Conference Spreadsheet Risk Management, Imperial College, London, pp. 121-136, ISBN: 978-1-905404-54-4

[30] Papp, P. & Csernoch, M. (2018) Spreadsheeting is Problem solving? In Hungarian: A táblázatkezelés is problémamegoldás? InfoÉra 2018. Retrieved 03. 06. 2020. from https://people.inf.elte.hu/szlavi/InfoDidact18/Infodidact2018.pdf?fbclid=Iw

AR38Vk3h2w_81Iv61C76V6xkErpLdxX4Ubc96P4VuR4EXYmIFW0b5-Jj0Z4

[31] Kirschner, P. A. & De Bruyckere, P. (2017) The myths of the digital native and the multitasker. Teaching and Teacher Education. 67 (2017), pp. 135-142

[32] Kirschner, P. A., Sweller, J., & Clark, R. E. (2006) Why Minimal Guidance During Instruction Does Not Work: An Analysis of the Failure of Constructivist Discovery, Problem-Based, Experiential, and Inquiry-Based Teaching. Educational Psychologist, 41(2), pp. 75-86

[33] Prensky, M. (2001) Digital Natives, Digital Immigrants. On the Horizon (MCB University Press, Vol. 9, No. 5, October 2001), Retrieved 03. 06. 2020. from http://www.marcprensky.com/writing/Prensky%20-%20Digital%20Natives,%20Digital%20Immigrants%20-%20Part1.pdf

[34] Champagne, A. B., Gunstone, R. F., & Klopfer, L. E. (1983) Naive knowledge and science learning. Research in Science and Technological Education, 1983, 1(2), pp. 173-183

[35] Csernoch, M. & Biró, P. (2017) Are digital natives spreadsheet natives? European Spreadsheet Risk Interest Group Conference, EuSpRIG2017, Imperial College, London. https://arxiv.org/abs/1909.00865, 2019

[36] Sestoft, P. (2011) Spreadsheet technology, IT University of Copenhagen, December 2011

[37] Burnett, M. M. & Scaffidi, C. (2012) End-User Development. The Encyclopedia of Human-Computer Interaction, 2nd Ed., https://www.interaction-design.org/literature/book/the-encyclopedia-of-human-computer-interaction-2nd-ed/end-user-development, 2012

[38] Csernoch, M. & Biró, P. (2015) Sprego programming. Spreadsheets in Education (eJSiE) (8) 1. Retrieved 03. 06. 2020. from https://sie.scholasticahq.com/article/4638-sprego-programming

[39] Booth, S. (1992) Learning to Program: A Phenomenographic Perspective. Goteborg Studies in Educational Sciences 89. Acta Univesitatis Gothoburgenis, Gothenbur

[40] Chmielewska, K. & Gilányi, A. (2019) Computer Assisted Activating Methods in Education. 2019 10th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), 2019, pp. 241-246, doi: 10.1109/CogInfoCom47531.2019.9089900

[41] Biró, P. & Csernoch, M. (2015) The mathability of computer problem solving approaches. 2015 6th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), 2015, pp. 111-114, doi: 10.1109/CogInfoCom.2015.7390574

[42] OFI (2013l) Frame Curricula for primary school grade 5-8. Compulsory subjects 1.2.3. In Hungarian: Kerettanterv az általános iskola 5-8. évfolyamára. Kötelező tantárgyak 1.2.3. Retrieved 03. 06. 2020. from http://kerettanterv.ofi.hu/01_melleklet_1-4/1.2.3_matemat_1-4_u.docx

[43] OFI (2013m) Frame Curricula for primary school grade 5-8. Compulsory subjects 2.2.0.3. In Hungarian: Kerettanterv az általános iskola 5-8. évfolyamára. Kötelező tantárgyak 2.2.0.3 Retrieved 03. 06. 2020. from http://kerettanterv.ofi.hu/02_melleklet_5-8/2.2.03_matemat_5-8_u.docx

[44] Biró, P. & Csernoch, M. (2015) The mathability of spreadsheet tools. 2015 6th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), 2015, pp. 105-110, doi: 10.1109/CogInfoCom.2015.7390573

[45] Chmielewska, K. & Matuszak, D. (2017) Mathability and coaching. 2017 8th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), 2017, pp. 427-432, doi: 10.1109/CogInfoCom.2017.8268284

[46] Chmielewska, K., Gilányi, A., & Łukasiewicz, A. (2016) Mathability and Mathematical Cognition. 2016 7th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), 2016, pp. 245-250, doi: 10.1109/CogInfoCom.2016.7804556

[47] Chmielewska, K. & Gilanyi, A. (2015) Mathability and computer aided mathematical education. 2015 6th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), 2015, pp. 473-477, doi: 10.1109/CogInfoCom.2015.7390639

Error Recognition Model: High-mathability End-user Text Management

Sebestyén Katalin[a], Csapó Gábor[a], Mária Csernoch[b], Bernadett Aradi[b]

[a] Doctoral School of Informatics, University of Debrecen
Kassai út 26, 4028 Debrecen, Hungary
sebestyen.katalin@inf.unideb.hu, csapo.gabor@inf.unideb.hu

[b] Faculty of Informatics, University of Debrecen
Kassai út 26, 4028 Debrecen, Hungary
csernoch.maria@inf.unideb.hu, aradi.bernadett@inf.unideb.hu

Abstract: Discussion, evaluation and error recognition, in natural language digital texts, is one of the most neglected areas in the digital world, despite the fact that text management is the most prevalent computer related activity. Millions of erroneous text-based documents of different types are in circulation, without us being aware of how fragile, damaged and harmful they are. It is well accepted in programming and even in other end-user activities, that error recognition plays a crucial role in teaching, learning and in real-world problem-solving processes. In the present paper, we introduce the High-mathability Error Recognition Model, which consists of the processes used in discussion and concept-based problem-solving and we also provide examples of the utilization of the model. We argue that error recognition and correction, and the assessment of problems in text management are as important as in other fields of informatics and computer sciences. In our study experimental groups – studying with the Error Recognition Model – and control groups – studying with low-mathability tool-centered approaches – were compared. It was found that the Error Recognition Model is more effective in digital text management, than for the tool-centered methodologies, in two error types: the typographic and layout-breaking error categories and a strong compensation effect was found in the syntax error category.

Keywords: teaching/learning strategies; lifelong learning; improving classroom teaching; data science applications in education; information literacy

1 Introduction

A well-accepted and effective method in teaching programming is to present erroneous source codes for students to detect and correct mistakes and errors [1] [2] [3]. This method is used to teach students program-formulation and language-constructs [4] and to prepare them for searching for errors, either in their own or in others' programs. However, program codes are not the only source of errors in the digital world. Natural language texts are more error-filled [5] [6] [7] [8] [9] [10] [11] [12] [13], a problem which has various reasons: the wide variety of languages, the grammar of languages – which is more complex than that of artificial languages –, and the level of users' knowledge (none do programming on a daily basis, without any background knowledge). Beyond the special features of languages, a text should meet several other requirements. The contents of the texts – their creation with the attendant typing difficulties or from copying, not to mention any copyright issues, and their language settings and spelling and grammar checkers – also raise considerable issues. Among these requirements, typographic [14] [15] [16] [17] [18] rules play a crucial role, with an emphasis on the various interfaces used to display content, and how to make the content the most legible on the selected interface.

However, because word processers are available for all, we are faced with several misconceptions regarding end-user text management, which leads to widespread 'bricolage' [5] [6] [7] [8] [9] [10] [11] [12] [13]. The first problem is that end-users are untrained or self-trained, and, compared to programmers, most of them are never formally taught how to design a text, how to build an algorithm for the text-management process, and how to format properly [24] [28]. They do not even know what the requirements of a properly formatted text are [12] [13]. They are in the comfortable state of not knowing what they do not know, the Dunning-Kruger effect [19].

The second problem is that end-user text management, in most cases is identified as interface-navigation – wrapped up in the 'user-friendly' slogans of the software companies – which leads to low-mathability activities [7] [29] [30] [31]. Consequently, nothing is learned, nothing is stored in long term memory, and no schemata are constructed, which leads to the intensive use of attention mode thinking, which is more prone to produce errors than automatic mode thinking. It is not surprising that some researchers find text and spreadsheet management low level, boring, routine activities [20] [21], in spite of the warnings that end-user computing should be taken seriously [22] [32]. This discrimination against end-user computing can also be explained by teachers' belief in the "fixed" nature of the subject and their own low-level efficiency [23].

1.1 Text Management – a Problem Solving Approach

In end-user text management, the most common document types are digital texts (texts handled with word processors and text editors), presentations, and web pages. Within this framework, the text management process follows the four steps of Pólya's concept-based problem-solving approach [24] [12] [25] [27]. In our context, the focus is on the discussion element – i.e. on evaluation, error recognition, classification and correction, which is ever-present in natural language text management, and which therefore has a great impact on the whole process and the output(s). In the present paper we introduce the Error Recognition Model and detail a test to compare the effectiveness of the traditional, surface approach methods and the Error Recognition Model.

The concept-based problem solving approach of Pólya [24], recently recognized as a high-mathability problem solving method [29] [30] [31] [33] [34] [35] [36] [37], has proved efficient both in maths and programming (and many other subjects); the different phases of this approach are recognized as the levels of mastery in the digital age [25], and have been found to be adaptable to the other popular end-user activity, such as spreadsheet management [38] [39].

Pedagogical Content Knowledge (PCK) [40] [41] and/or Technological Pedagogical Content Knowledge (TPCK) [39] are required for effective teaching in the digital era. Most teachers apply low mathability approaches to computer problem solving; teaching their students with a fixed belief in science – IT, ICT, CS – and with low self-efficacy. These approaches mainly focus on interface navigation, and passing exams, without providing students opportunities to fully understand and appreciate science [23] and can lead to erroneous end-user activities [5] [6] [7] [8] [9] [10] [11] [12] [14] [13] [22]. As mentioned above, effective end-user text management requires the cooperation of teachers of different subjects. Consequently, teacher education should be changed according to the requirements of high mathability problem solving; this would be preferable to teaching how to use a software environment in pre- and in-service courses. A similar approach was suggested in spreadsheet management by Angeli in 2013 [39], but the idea has not reached the wider public, and remains isolated. Crowd sourcing would help us reach a wide range of teachers who would understand that end-user text management involves lot more than typing, and copying – not infrequently without naming sources – and clicking on the buttons of the toolbars offered by the software companies [42].

Crowd sourcing would also help find IT professionals who at present, mostly ignore end-user computing [22] [43] and could recognize that High Mathability End-user Computing would be an effective introduction to serious computing.

1.2 The Error Recognition Model (ERM)

Due to the extremely high number of possible errors in a digital natural language text, it is necessary to classify them. The main classes of errors are constructed on the systems of rules which create the major guidelines for a correct natural language text. A digital text should fulfill the requirements of the language(s) and the content of the text, as well as the rules relating to displaying, breaking or layout, and formatting. These are the most common rule systems which should be taken into consideration in the process of end-user text management. Violation of these rules leads to errors of the following types: syntactic, semantic, typographic, layout or breaking, formatting, and style errors [12] [27] [28].

2 Hypotheses

Based on the theoretical background of the high-mathability Error Recognition Model in text-management, we launched a research project, whose aim was to prove the effectiveness of the approach compared to the interfaces centered, traditional methods. Five hypotheses are formulated to see how the error recognition abilities of students develop using the traditional and the newly introduced ERM model. To complete our study, a testing series was administered in primary and secondary schools within the topic of word processing and text-management.

[H1] In the pre-test there is no difference between grade 9 experiment (9E) and control groups (9C).

[H2] In the pre-test there is a difference between grade 7 and grade 9 students.

[H3] In the post-test there is a difference between grade 9 experiment (9E) and control groups (9C).

[H4] In the post-test there is a difference between grade 7 experiment (7E) and grade 9 experiment groups (9E).

[H5] In the post-test there is a difference between grade 7 experiment (7E) and grade 9 control groups (9C).

3 Testing

To quantify and prove the efficiency of the ERM method, we tested experiment groups where this novel, high-mathability [28] [29] [30] [31] [33] [34] [35] [37] approach was introduced, and compared their results to control groups where the traditional, low-mathability, tool-focused methods are used.

3.1 Sample

The teaching and testing process took place in the academic year 2017/2018, in one of the high schools in Debrecen, Hungary. Students from grade 7 and 9, formed both the experimental and control groups (Table 1). Considering the background knowledge of the participating students, the selection of groups plays a crucial role, because all of the students had learned the basics of word processing – whatever the word “basic” means in this context – in primary education. In general, according to the Hungarian frame curricula [44] [45], these students are able to construct text documents based on a sample provided [45]. The methods with which students were taught in primary education are not documented; however, the structure of the frame curricula and the structure and content of the textbooks clearly indicate that primarily the traditional, tool-focused methods are applied.

During our experiment, students were tested in two rounds: in a pre- and a post-test. The pre-tests were administered in advance of the teaching-learning process, to register what knowledge the students bring with them from their previous studies, while the post-tests were completed at the end of the topic, both groups.

Table 1
The number of students who participated in both tests

	Experimental groups	Control groups	Total
Grade 7	26	–	26
Grade 9	66	38	104
Total	92	38	130

Considering all groups, 153 students completed the pre-test. 34 and 69 students participated in the experiment from grades 7 and 9, respectively. 146 students completed the post-test: 102 in the experiment and 44 in the control groups. Pairing the students, 130 students completed both the pre- and post-tests, 92 from the experiment groups and 38 from the control groups (Table 1).

3.2 Conducting the Measurements

The students had 45 minutes to complete the tests in both the pre- and post-tests. Both tests consist of three phases:

(1) Unplugged

(2) Semi-unplugged error recognition

(3) Plugged-in error correction (not detailed in this paper)

During the unplugged phase, the computers were turned off and each student got a printed version of the text document and a blue pen. The students' task was to scan the printed document and mark (circle) and name the errors or error types which they identify. We did not expect the students to provide the terminologically correct name for each error, but to give a short explanation of why they marked that part of the document as erroneous.

In the next phase we collected the blue pens and the students opened the electronic version of the document and got a red pen to mark and explain on the paper the errors they discover in the digital version of the document.

The third and last phase of the test was to correct the errors and to save the document using a name and folder provided. In the present study the results of the recognition of errors and the analyses of these tasks are presented. We also must note that recognizing the semantic errors in the documents is beyond the scope of this analysis; consequently, any further details are not provided regarding this category of errors.

The process of testing and the evaluation of the pre- and post-tests are identical. However, a larger amount of text and more complex errors were presented in the post-test. The length and quality of the document used in the post-test adjusts to the acquired level of knowledge in the topic and is matched with the students' time management ability. The differences between the documents selected for the pre- and post-tests originated from our partially different goals. In the pre-test we were interested in documenting the knowledge the students brought with them from their previous studies, while in the post-test, we wanted to register the students' improvement in applying the different approaches in the teaching-learning process.

3.3 Errors in the Test Documents

In Table 2, the errors, the error types, and the place of recognition of the pre-test are listed. In Table 3 we present the errors of the post-test and also in the Error column we mark which errors occur in both tests (PE/PO) or only in the post-test (PO), where PE refers to the pre-test and PO to the post-test.

Table 2

Errors in the pre-test, categorized by error types

Error type	Error	Place
syntactic	– spelling mistakes – improper use of parentheses with Space characters	printed
typographic	– underline – whole text italic	printed
layout-breaking	– empty paragraphs – paragraph marks at the end of each line – indentation with Space characters – alignment with Space characters – manual hyphenation	digital

Table 3

Errors in the post-test, categorized by error types. Errors marked PE/PO are present in both the pre-test and the post-test, while errors marked PO are only present in the post-test.

Error type	Error	Place
syntactic	– spelling mistakes (PE/PO) – improper use of parentheses with Space characters (PE/PO) – missing Space characters (PO)	printed
typographic	– underline (PE/PO) – italic (PE/PO) – bold (PO) – all capitals (PO)	printed

layout-breaking	– empty paragraphs (PE/PO) – paragraph marks at the end of each line (PE/PO) – indentation with Space characters (PE/PO) – alignment with Space characters (PE/PO) – manual hyphenation (PE/PO) – manual numbering (PO) – multiple Space characters (PO) – character spacing, expanded (PO)	digital

Students got points if they recognized – marked –, named, and categorized the errors. In the pre-test, there are 7 syntactic, 2 typographic, and 5 different layout-breaking errors of varying appearance. In the post-test we can recognize 10 syntactic, 4 typographic, and 8 layout-breaking errors, where in all three categories multiple occurrences are detectable. We must note here that in the case of repeated instances of the same errors, on recognizing more than half of the same errors, students were awarded additional points.

4 Methods

4.1 The Evaluation Process

Following the administration of the tests, the evaluation process took place. In advance of the actual evaluation process an evaluation table was set up in Excel, where all the items were listed.

The items of the tests were decided based on the nature and the frequency of the errors (Table 4). The following items of the evaluation table served as the basis for the statistical analyses:

- Primarily the errors were grouped on the basis of the three error categories (Table 2 and Table 3): syntactic, typographic, and layout-breaking.
- Within the categories all the errors, were checked according to the previously established smaller items.
  - The errors had to be marked on the paper and the also had to be named (markers without any written notes were not accepted as correct answers).
  - The color of the markers and/or the notes were also checked to reveal the place of recognition. The syntactic and the typographic errors are recognizable on the hard copy, while the layout-breaking errors only in the digital form of the document.
  - In the case of multiple errors, we also checked and recorded how many of the same error were recognized. If more than half of the same errors

were marked, an additional point was added to the sum. For example, in the pretest, three additional points were available for marking multiple space characters, paragraph marks imitating vertical spacing, and paragraph marks at the end of lines.

Table 4

The items of the three errors types and their relative frequency in the three error categories in the pre- and post-tests

	Pre-test		Post-test	
	number of items	rel. frequency	number of items	rel. frequency
syntactic	13	43.33%	37	53.62%
typographic	4	13.33%	12	17.39%
layout	13	43.33%	20	28.99%
TOTAL	**30**	**100.00%**	**69**	**100.00%**

We must note here that in the pre-test there is a formatting error, where one of the paragraphs is formatted with Keep with next; however, none of the students recognized it. Furthermore, this type of error is not included in the post-test; consequently, we cannot examine how students developed. In general, this error is omitted from the analyses.

The Statistical Analyses

Considering our hypotheses, we focused on the following test groups / comparisons:

- Grade 9 experimental groups (9E) versus grade 9 control groups (9C)
- Grade 9 experimental groups (9E) versus grade 7 experimental groups (7E)
- Grade 9 control groups (9C) versus grade 7 experimental groups (7E)
- Pre-tests versus post-tests

The statistical analyses were carried out with the following methods.

First, to check whether the samples follow normal distribution or not, we used the 1-sample Kolmogorov-Smirnov test. Here the null hypothesis is that the sample is drawn from the reference distribution (normal, in our case).

Because in many cases we found that the samples we want to compare do not follow normal distributions, we used the nonparametric Mann-Whitney U test to compare the samples. The null hypothesis in this case is that the medians of the two samples are the same. We used this test for independent samples.

However, in some cases the normality assumption of Student's t-test was satisfied. In these situations, we used the latter test to compare the means of independent samples. Sometimes we could assume that the variances are equal, sometimes not; we checked this condition with Levene's test. To perform the statistical tests, we used SPSS.

5 Results

5.1 Pre-test

The comparison of the students' results was based on the three error types – syntactic, typographic, and layout-braking categories – and the errors listed in section 3.3 (Table 2 and Table 3). In the pre-test, considering the total points, the students achieved 20.29%, 31.52%, and 23.67% in the 7E, 9E, and 9C groups, respectively (Table 5). Furthermore, we checked for significant differences between three groups comparing the three error categories. The grades were compared in all the possible variations:

The Kolmogorov-Smirnov Test proved that the results of the pre-tests in the three error categories do not necessarily follow normal distribution. All but one of the three error categories and the three groups of students showed a non-normal distribution:

- 7E: syntax, typography, layout-breaking
- 9E: syntax, typography, layout-breaking
- 9C: typography, layout-breaking

Consequently, we used the nonparametric Mann-Whitney U test to compare the pretest results.

In the typographic errors category all students in all three groups scored 0 points; therefore, there is no difference, so we only checked the other two error categories.

According to the Hungarian National Base Curriculum [44] and the frame curricula [45], grade 9 students had studied word processing in their previous studies; consequently, we assumed that in the pre-test no significant difference would be recognized between the two groups [H1]. Considering the total results, in the comparison of grade 9 students, no difference was found between the experiment and control groups (p=0.103). However, in the layout-breaking category the result of the experiment group was significantly higher than the control group (p=0.014). Considering the syntactical errors, the 9E groups scored higher, with no significant difference between the two groups (p=0.879) (Table 5). This latter result is in accordance with the 9E and 9C groups acceptance results for high school, however we must emphasis that in the totals there is no significant difference between the grade 9 groups [H1].

In the comparison of grade 7 and grade 9 students, based on their previous studies according to the NAT [44] and the frame curricula [45], we expected significant differences between the age groups [H2]. When comparing grade 9E with 7E, in accordance with our [H2] hypothesis, we found a significant difference in the total results (p=0.003) and in the layout-breaking error category (p=0.000). However, no significant difference was found in the syntax category (p=0.482). On the contrary, between the 9C group and the 7E groups, considering their total results, no significant difference was found (p=0.401). In the comparison of the two error groups, neither the syntax (p=0.505) nor the layout-breaking (0.083) error groups showed a significant difference. Based on the results of the pre-test, we can neither confirm nor reject our [H2] hypothesis. This leads to the conclusion that previous studies in word processing and text-management do not necessarily build up long lasting, firm, reliable knowledge.

Table 5

The results (%) of the experiment and the control groups in the pre-test regarding the three error categories

	7E	9E	9C
Syntactic	40.72%	43.59%	41.08%
Typographic	0.00%	0.00%	0.00%
Layout-breaking	6.11%	29.14%	13.54%
TOTAL	**20.29%**	**31.52%**	**23.67%**

Considering syntactical errors, there is no significant difference between the two age groups; therefore, the results are independent of age and the previous experiment. In the case of syntactic errors, we can conclude that previous studies either in informatics or native language did not help students in recognizing printed errors of this type. As mentioned above, in the typographic category all students scored 0, which proves that previous studies did not pay attention to the typographic rules regarding the printed version of the documents.

5.2 Post-test

The comparison of the results of the experiment and control groups in the post-test was conducted to reveal the differences between the two teaching-learning approaches: the traditional vs. the ERM method. The average scores of the three groups of students in the three categories are presented in Table 6.

Table 6

The average results (%) of the three groups of students in the post-test

	7E	9E	9C
Syntactic	19.01%	26.40%	35.75%
Typographic	67.53%	60.05%	1.14%
Layout	41.72%	49.11%	35.68%
TOTAL	**34.03%**	**38.83%**	**29.71%**

In the case of the syntactic errors the 9C groups produced better results than both the experiment groups. The 7E and 9E groups reached 19.01% and 26.40%, respectively, while 9C achieved 35.75%. The difference between the groups are significant in all cases (9E vs. 9C p=0.002; 7E vs. 9E p=0.048; 7E vs. 9C p=0.000).

Considering typographic errors in the post-test, the experiment groups (9E vs. 9C p=0.000; 7E vs. 9E p=0.172; 7E vs 9C p=0.000) proved to be significantly better than the control groups (p=0.000). This result clearly demonstrates that the typographic problems were not handled in the control groups, while in the experiment groups the subject was covered. The results of the post-test clearly reveal that students of the experiment groups learned the fundamental typographic rules, which play an important role in the presentation of any text-based document. In the comparison of the two age groups of the experiment groups, the results of grade 7 students (67.53%) were somewhat higher than those of grade 9 students (60.05%) (Table 6), but there is no significant difference between the two age groups (p=0.172).

In the layout-breaking error category, it was found that grade 9E groups achieved the highest results (49.11%), followed by the 7E groups (41.72%); the 9C groups scored the lowest (35.68%) (Table 6). A significant difference was found between the 9E and 9C groups (p=0.000). However, between the 7E and 9E groups (p=0.450) and the 7E and 9C groups (p=0.145) no significant difference was found.

Summarizing the results of the post-test, we found that in two error types – typographic and layout – the experiment groups, studying with the Error Recognition Model, provided better results. In both cases we have found that the ERM model applied in the teaching-learning process gives at least a two year-advantage compared to the traditional, interface-focused, low-mathability approaches.

In the category of syntactic errors, we have revealed a different pattern. Most of the items in the syntactic category were grammatical errors, i.e. knowledge which is, officially, acquired in native language classes. In the process of handling text-based documents knowledge built up in another school subject has to be transferred to the digital environment. As mentioned above, there was significant difference between the 9E and 9C groups. Considering these results, it seems that the 9C group compensates for their lack of knowledge with syntax, in other word, they primarily focus on this error category.

However, at this point we must call attention to the low percentage of students recognizing grammatical errors, mainly spelling errors. This knowledge comes from native language classes, where students are supposed to write without grammatical errors at this age. We can conclude that the knowledge transfer rate between of native language and informatics subjects is extremely low. Consequently, we must develop a higher level of cooperation between the teachers of the two subjects, so that students can apply their grammatical knowledge in digital environments.

These findings prove hypothesis [H3], but we must evaluate the results by error categories. In the typographic and layout-breaking errors the ERM model is clearly more effective than the traditional method. However, the compensation in the syntax category, despite its low scores, resulted in higher scores in group 9C.

In general, the ERM group recognize more types of errors, but these new errors distract them from the syntactical errors. It seems that the students' working memory is flooded and they need more schemata to be able to handle all the errors effectively.

The comparison of the grade 7 and 9 experiment groups revealed that in typographic and layout-breaking categories, the two groups scored similarly, with no significant difference between them. Considering the syntactical errors, the older students' results were significantly higher. This proves that the capacity of the working memory of the younger students is less than that of the older students, who might bring in schemata built up in previous studies, either in informatics or native language (Hungarian) classes. Consequently, hypothesis [H4] is rejected when the ERM method is used. The results of the pre-test prove that the selection of method is crucial, because hypothesis [H2] was only partially accepted.

In hypothesis [H5], we assumed that there is difference between groups 7E and 9C. Our analysis proved that apart from the syntax compensation, the younger students scored higher in both typography and layout-breaking errors, with significant and no significant differences, respectively. In the comparison of hypotheses [H3] and [H5] the ERM is more effective than the traditional method, even overriding the age differences.

5.3 The Comparison of the Results of the Pre- and Post-Tests

In Table 1 the number of students participating in both tests, and the way their numbers are distributed between the experiment and control groups and age groups are presented. We worked with, and used for comparison, the results expressed in percentage format due to the different number of items in the test (Table 4).

Table 7
The comparison of the results of the pre- and post-tests

	7E		**9E**		**9C**	
	Pre-test	Post-test	Pre-test	Post-test	Pre-test	Post-test
Syntactic	41.12%	19.75%	44.87%	27.02%	42.11%	36.27%
Typographic	0.00%	67.95%	0.00%	60.61%	0.00%	0.88%
Layout	4.14%	41.73%	30.65%	49.32%	12.55%	36.58%
TOTAL	**19.62%**	**39.68%**	**32.73%**	**45.23%**	**23.68%**	**34.74%**

In the pre- and post-tests both the experiment and control groups improved in recognizing and marking two of the error categories: typographic and layout-breaking errors (Table 7, Figure 2 and Figure 3). Considering syntactic errors, all the groups' results were lower in the post-test than in the pre-test. However, the difference in the 7E ($p=0.000$) and 9E ($p=0.000$) groups was significant, while in the 9C group it was not ($p=0.102$). This results also proved that the control group focuses only on syntax errors, leaving no place for typography in the unplugged phase.

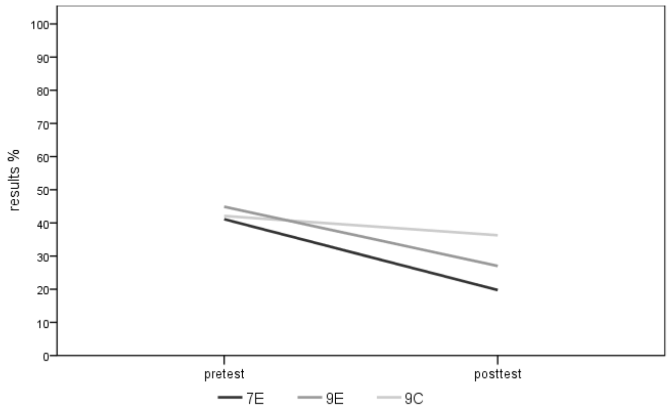


Figure 1
The results in the pre- and post-test, regarding syntax errors

The comparison of the pre- and post-test proved that the recognition of syntactic errors greatly depends on the knowledge built up in native language classes and bought into informatics. The short text of the pre-test, with a relative frequency of 43.33% for syntactical errors, suited 7E students better than the longer text of the post-test. One further explanation would be that while in the pre-test the students focused primarily on syntactical errors, in the post-test they divided their attention between the different error groups. However, this assumption requires further research. In general, we can conclude that the time spent on text management in informatics classes is not enough to transfer grammatical knowledge from other classes. One solution would be that language classes apply digital tools for integrating language knowledge into digital texts.

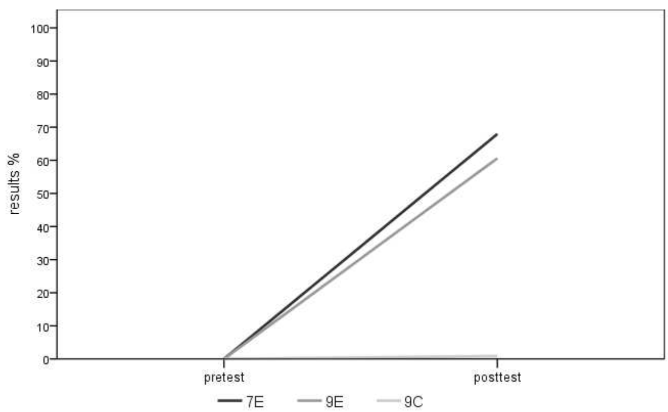


Figure 2
The results in the pre- and post-test, regarding typographic errors

In the typographic error category the experiment groups underwent a strong and significant improvement (7E, 9E: p=0.000), while the control group stagnated and produced similar results in both tests with no significant difference between the two tests (p=0.160) (Figure 2).

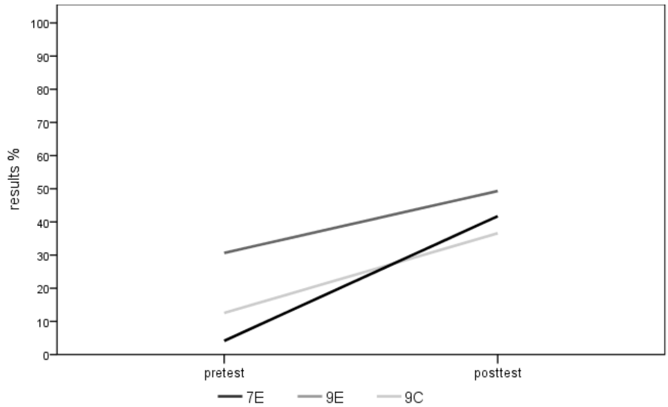


Figure 3
The results in the pre- and post-test, regarding layout-breaking errors

All three groups show a significant improvement in the layout-breaking error type (7E, 9E: p=0.000, 9C: p=0.000). Not only was the development between the pre- and post-tests significant, but the rate of improvement between the participating groups (experiment and control) was also noteworthy (p=0.000) (Figure 3). In text management, recognizing and naming errors all students showed an improvement; however, students working with the ERM achieved significantly better results compared to the students who learned with traditional approaches.

5.4 The Rate of Development

We were interested to see how students improved during this special teaching-learning period. Considering the syntax error group, it was found that all the groups' results were lower in the post-test than in the pre-test. On the other hand, the recognition of the layout-breaking errors improved in all the three groups, as did the recognition of the typographic errors in the two experiment groups. It seems that simultaneously, with the development of the students, a switch in focus is recognizable; while in the pre-test their knowledge was restricted to syntax, in the post-test their knowledge space was widened. In the experiment groups the two other error categories are taken care of, while in the control group only one category is, which explains the relatively better syntax results of the control group.

In syntax, the greatest drop is seen in the 7E group. There is no significant difference between the two experiment groups in terms of this drop (p=0.545), although there is a significant difference between the experiment and control groups: groups 7E and 9C (p=0.009) and groups 9E and 9C (p=0.016) (Table 7). Figure 3 clearly shows these results with the corresponding parallels for the two experiment groups. Again, proof was found for the compensation for the lack of other knowledge in group 9C.

The development of the three groups in terms of typography showed different patterns. In this respect, the development of the 9C group is significantly lower than that of the other two groups (p=0.000, p=0.000). However, there is no significant difference between the development of the two experiment groups (p=0.264). In general, the greatest development was registered in the 7E group.

The pattern of development of the layout-breaking errors is different from the other two error categories. It was found that there was no difference between groups 9E and 9C in the development of this error type (p=0.173), while there is significant difference between the two experiment groups (p=0.000), and groups 7E and 9C (p=0.003). This result is also clearly shown in Figure 4 which shows the parallel results for the two grade 9 groups.

6 Summary

The results clearly show that either the ERM or the traditional method do not have a direct influence on the development of the ability to recognize syntactical errors. Furthermore, we have found that the perspective of the experiment groups widens; consequently, they can divide their attention between the three error categories, while the control group can only handle two of the categories. In this case, the 9C group in the unplugged phase compensates for the typographic errors with the syntax. The greatest development was recorded in the typographic errors in the experiment groups. In the layout-breaking errors, all the groups developed significantly.

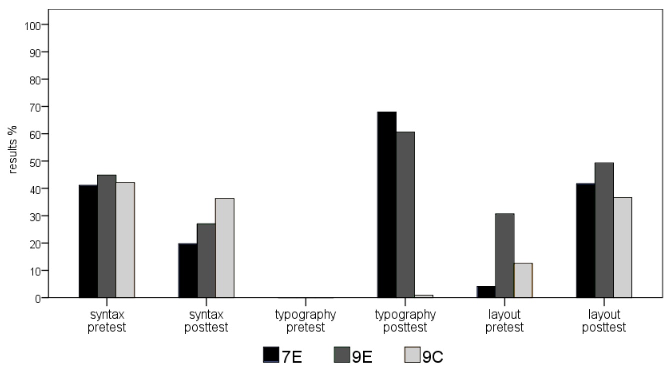


Figure 4

The results of the pre- and post-tests in groups 7E, 9E, and 9C

In general, (Figure 6), the 9E group's result was the highest, but the 7E group's development was the greatest. Here, based on our results, we can conclude that both subjects – recognition of typography and layout-breaking errors – can be taught in middle schools as effectively as in high schools. We would suggest starting these subjects as soon as possible, as students are ready for, and receptive to, this knowledge.

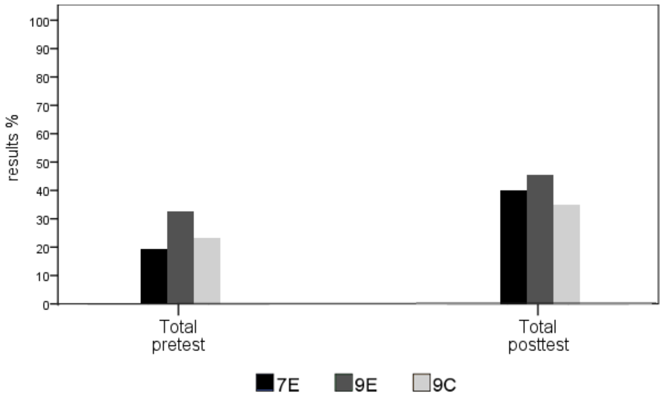


Figure 5
The total results of the pre- and post-tests in groups 7E, 9E, and 9C

Conclusions

In the present research, the effectiveness of two teaching methods, connected to word processing and digital text management, were analyzed. The traditional, widely accepted and supported surface approach methods, focusing on the features of word processors were compared to the newly introduced Error Recognition Model (ERM).

We tested three groups of students – grade 7 experiment (7E), grade 9 experiment (9E), and grade 9 control (9C) – with a pre- and a post-test.

We found that the Error Recognition Model, introduced in the experiment groups proved to be more effective in digital text management than the traditional methodologies in two error types: the typographic and layout-breaking error categories, while a strong compensation effect was found in the syntax error category. The compensation in this context means that if students are familiar with several error groups, they divide their attention among them. However, the fewer error categories they know, the more their primary focus is on syntactical errors. This finding was proved twice in our experiment: (1) In the pre-test, when the typographic and layout-breaking errors scored low, but the syntactical errors relatively high. This result was reversed in the experiment group in the post-test. (2) The control group could not improve significantly in the recognition of either the typographic or layout-breaking errors, but only in the detection of the syntactical errors.

In our opinion, this compensation effect is strongly related to the capacity of working memory and the schemata built up in previous studies. However, this latter finding requires more research and comparison.

The method based on the Error Recognition Model is a complex, time-consuming process, which assumes cooperative work from teachers and periodical assessment from both teachers and students, in order to reach a state which fulfills the requirements of digital text documents.

Acknowledgements

This work was supported by the construction EFOP-3.6.3-VEKOP-16-2017-00002. The project was supported by the European Union, co-financed by the European Social Fund.

References

[1] Gould, J., Drongowski, P.: An exploratory study of computer program debugging. Human Factors, 16, 1974, pp. 258-277

[2] Gould, J.: Some psychological evidence on how people debug computer programs. International Journal of Man-Machine Studies, (7) 1, 1974, pp. 151-182

[3] Jerinic, L.: Teaching Introductory Programming Agent-based Approach with Pedagogical Patterns for Learning by Mistake. (IJACSA) International Journal of Advanced Computer Science and Applications, 2014

[4] Chan Mow, I. T.: Analyses of Student Programming Errors In Java Programming Courses. Journal of Emerging Trends in Computing and Information Sciences, 2012, (3)5

[5] Ben-Ari, M.: Bricolage Forever! PPIG 1999. 11th Annual Workshop. 5-7 January 1999. Computer-Based Learning Unit, University of Leeds, UK. Retrieved: 01/03/2020 from http://www.ppig.org/sites/ppig.org/files/1999-PPIG-11th-benari_0.pdf

[6] Ben-Ari, M., Yeshno, T.: Conceptual models of software artifacts. Interacting with Computers, 18(6), 2006, pp. 1336-1350

[7] Biró, P., Csernoch, M.: The mathability of computer problem solving approaches. 6th CogInfoCom, Győr, 2015, pp. 111-114

[8] Csernoch, M.: Methodological Questions of Teaching Word Processing. 3rd International Conference on Applied Informatics, Eger-Noszvaj, Hungary, 1997, pp. 375-382

[9] Csernoch, M.: Teaching word processing – the theory behind. Teaching Mathematics and Computer Science, 2009, 2009/1, pp. 119-137

[10] Csernoch, M.: Teaching word processing – the practice. Teaching Mathematics and Computer Science, (8)2, 2010, pp. 247-262

[11] Csernoch, M.: Clearing Up Misconceptions About Teaching Text Editing. In: I Candel Torres L Gómez Chova A López Martínez (ed.) ICERI2011: 4th International Conference of Education, Research and Innovation. Madrid, Spain, (IATED), 2011, pp. 407-415

[12] Csernoch, M.: Do You Speak and Write in Informatics? (2019) The 10th International Multi-Conference on Complexity, Informatics and Cybernetics, March 12-15, 2019, Orlando, Florida, USA, 2019, pp. 147-152

[13] Csernoch, M., Bujdosó, Gy.: Errors of exams and competions in informatics and their consequences. In Hungarian: Vizsga- és versenyfeladatok szövegbeviteli hibái és ezek következményei, Új Pedagógia Szemle 1, 2009, pp. 19-40

[14] Virágvölgyi, P.: The mastery of typography with computers. In Hungarian: A tipográfia mestersége számítógéppel, Osiris Kiadó Kft., 2004

[15] Jury, D.: About Face: Reviving The Rules Of Typography, RotoVision SA, Switzerland, 2004

[16] Jury, D.: What is Typography?, RotoVision SA, Switzerland, 2006

[17] Reynolds, G.: Presentation Zen: Simple Ideas on Presentation Design and Delivery, Pearson Education Inc., 2008

[18] Bujdosó, Gy., Csernoch, M.: Digital literacy, digital grammar. In Hungarian: Digitális írástudás, digitális nyelvhelyesség. Tudományos és Műszaki Tájékoztatás, 61(10), 2014, pp. 1-10

[19] Kruger, J., Dunning, D.: Unskilled and Unaware of It: How Difficulties in Recognizing One's Own Incompetence Lead to Inflated Self-Assessments. Journal of Personality and Social Psychology 77(6), 1999, pp. 1121-34

[20] Bell, T., Newton, H.: Unplugging Computer Science. Improving Computer Science Education. (Eds.) Kadijevich, D. M. Angeli, C. and Schulte, C., Routledge, 2013

[21] Gove, M.: Michael Gove speech at the BETT Show 2012. Published 13 January 2012. Digital literacy campaign. Retrieved 12/02/2020 from http://www.theguardian.com/education/2012/jan/11/digital-literacy-michael-gove-speech

[22] Panko, R., Port, D.: End User Computing: The Dark Matter (and Dark Energy) of Corporate It. Journal of Organizational and End User Computing, 25 (3), 2013, pp. 1-19

[23] Chen, J. A., Morris, D. B., Mansour, N.: Science Teachers' Beliefs. Perceptions of Efficacy and the Nature of Scientific Knowledge and Knowing. In International Handbook of Research on Teachers' Beliefs. (Eds.) Fives, H. & Gill, M. G. Routledge, 2015

[24] Pólya, G.: How To Solve It. A New Aspect of Mathematical Method. Second edition (1957) Princeton University Press, Princeton, New Jersey, 1954

[25] IEEE&ACM Report 2013: Computer Science Curricula 2013. Curriculum Guidelines for Undergraduate Degree Programs in Computer Science. December 20, 2013. The Joint Task Force on Computing Curricula Association for Computing Machinery (ACM) IEEE Computer Society. Retrieved 25/05/2019 https://www.acm.org/binaries/content/assets/education/cs2013_web_final.pdf

[26] Csernoch, M., Biró, P.: Teaching methods are erroneous: approaches which lead to erroneous end-user computing. EuSpRIG2016. London. Retrieved 02/03/2020 from http://www.eusprig.org/mcsernoch-2016.pdf

[27] Csernoch, M.: Algorithms and Schemata in Teaching Informatics. In Hungarian: Algoritmusok és sémák az informatika oktatásában II., 2016 Retrieved 25/01/2020 from http://tanarkepzes.unideb.hu/szaktarnet/kiadvanyok/algoritmusok_es_semak_2.pdf

[28] Csernoch, M., Biró, P.: Wasting Human and Computer Resources. International Journal of Social, Education, Economics and Management Engineering, 9(2), 2015, pp. 573-581

[29] Baranyi, P., Gilányi, A.: Mathability: Emulating and Enhancing Human Mathematical Capabilities. 4th CogInfoCom), 2013, pp. 555-558

[30] Baranyi, P., Csapo, A., Sallai, G.: Cognitive Infocommunications (CogInfoCom), Springer International Publishing Switzerland, 2015, p. 191, (978-3-319- 19607-7, http://www.springer.com/us/book/9783319196077#aboutBook

[31] Baranyi, P., Csapo, A.: Definition and Synergies of Cognitive Infocommunication, Acta Polytechnica Hungarica, 2012, Vol. 9, No. 1, pp. 67-83 (ISSN 1785-8860)

[32] Soloway, E.: Should we teach students to program? Communications of the ACM, 1993, 35(10) pp. 21-24

[33] Chmielewska, K., Gilányi, A.: Computer assisted activating methods in education. 10th CogInfoCom, Nápoly, 2019, pp. 241-246

[34] Chmielewska K., Gilányi, A.: Mathability and computer aided mathematical education. 6th CogInfoCom, Győr, 2015

[35] Chmielewska, K., Gilányi, A., Łukasiewicz, A.: Mathability and Mathematical Cognition. 7th CogInfoCom, Wrocław, 2016

[36] Chmielewska, K., Matuszak, D.: Mathability and coaching. 8th CogInfoCom, Debrecen, 2017

[37] Csernoch, M., Dani, E.: Data-structure validator: an application of the HY-DE model, 8th CogInfoCom, Debrecen 2018, pp. 197-202

[38] Hubwieser, P.: Functional Modeling in Secondary Schools using Spreadsheets. In Education and Information Technologies, 9(2), 2004, pp. 175-183

[39] Angeli, C.: Teaching Spreadsheets: A TPCK Perspective. Improving Computer Science Education. (Eds.) Djordje M. Kadijevich, Charoula Angeli, and Carsten Schulte. Routledge, 2013, pp. 132-145

[40] Shulman, L. S.: Those Who Understand: Knowledge Growth in Teaching. Educational Researcher, 15 (2), 1986, pp. 4-14

[41] Shulman, L. S.: Knowledge and Teaching. Foundations of the New Reform. Harvard Educational Review, 57, 1987, pp. 1-22

[42] Schulte, C., Saile, M.: Applying Standards to Computer Science Education. (Eds.) Djordje M. Kadijevich, Charoula Angeli, and Carsten Schulte. Routledge, 2013, pp. 117-131

[43] Kadijevich, D. M.: Learning about Spreadsheets. (Eds.) Djordje M. Kadijevich, Charoula Angeli, and Carsten Schulte. Routledge, 2013, pp. 19-33

[44] NAT 2012: National Base Curriculum. In Hungarian 110/2012. (VI. 4.) Korm. rendelete a Nemzeti alaptanterv kiadásáról, bevezetéséről és alkalmazásáról. Retrieved 12/02/2020 from http://ofi.hu/sites/default/files/attachments/mk_nat_20121.pdf

[45] OFI: Frame Curricula. In Hungarian: Kerettanterv. 51/2012. (XII. 21.) számú EMMI rendelet – a kerettantervek kiadásának és jóváhagyásának rendjéről. 2012, Retrieved 12/01/2020 from https://www.oktatas.hu/kozneveles/kerettantervek/2012_nat

Towards "Learnability"

What we benefit from Mathability

Katarzyna Chmielewska

Institute of Mathematics, Kazimierz Wielki University, Chodkiewicza 30, 85-064, Bydgoszcz, Poland, katarzyna.chmielewska@ukw.edu.pl

Abstract: Mathability in its definition, refers to cognitive infocommunication and combines machine and the human cognitive capabilities that are essential for mathematics. Through the years, the notion evolved and its methods turned to be applicable for self-education and constructive learning. In this paper, we would like to investigate how some of the methods can be practiced to meet requirements of the contemporary labor market, especially if we consider as an aid of ICT.

Keywords: Mathability; Education; Constructive Learning; Learning of Adults; Soft Skills; Activating Methods; Edu-coaching; Activating Educational Method; Life Long Learning; Self-improvement

Introduction

Contemporary labor market, called *Anytime, anywhere*, its dynamic changes as well as features of the Millennials and the next generation, make personal development methods more and more valuable already at an early stage of education. In this paper, we present aspects of constructive learning which we inherit from developing mathability theory. We will investigate how to apply the methods to achieve scientific success (in learning, solving problems, gaining mental abilities, etc.) and personal success (discovering talents and potential, changing attitude towards problems, communicating, cooperation in a group, management, etc.).

Taking into consideration young generations' habits (among others: [2] [18] [21] [41] [46]) it is evident that they learn, work, and entertain using ICT. Moreover, they build their social life in a virtual world. Current demand of life-long learning and self-development is supported by technology. The methods we will discuss serve both using the habits as sufficient and effective tools of self-improvement and enabling people to meet the employers' demands. All the facts are strictly related to the definition of cognitive infocommunications (CogInfoCom; cf. [1] [3] [4] [5] [48]) and contribute to cognitive sciences by matching cognitive process in education with infocommunications devices and methods.

The paper consists of four parts. First, we present an evolution of the theory of mathability. We explain how the notion has broadened its range. In the second part, we describe some features of the contemporary labor market, requirements of employers and some problems related to that. In the main part, we propose a way of solving these problems by adapting constructive methods typical for mathability development: in the third part we discuss a notion of Learnability and in the last part – we prove applicability of mathability methods.

1 Evolution of the Concept of Mathability

By *mathability* we used to understand human mathematical ability. In 2013 P. Baranyi and A. Gilányi introduced a broader idea of the concept (cf. [3] [5]). Mathability began to refer to a branch of Cognitive Infocommunications that investigates any combination of artificial and natural cognitive capabilities relevant to mathematics. The range of the notion varies from low-level arithmetic operations to symbolic reasoning of a high-level.

Among others, there were two main goals considered:

- To develop a set of methodologies using which human mathematical capabilities can be emulated and enhanced.
- To work out a measure with which artificial mathematical capabilities can be quantified.

1.1 Emulating Human Capabilities

Since 2014 P. Biró and M. Csernoch have investigated students’ ways of thinking while problem solving. Namely, they considered deep and surface metacognitive processes in non-traditional programming tasks, computer problem solving approaches and mathability of spreadsheet tools (read more in: [6] [7] [8] [27] [28]).

K. Bubno and V. L. Takács have considered application of word problem solving as initial computer programming exercises ([15]). They described usefulness of some methods originated from mathematical education to teaching computer science (16]). On the other hand, they presented how to solve mathematical problems using Blockly ([17]).

A. Gilányi and M. Merentes have investigated applicability of existing computer algebra systems for solving mathematical problems ([11] [12] [13] [33] [34]). They presented ways of getting ideas leading to discovering solutions of unsolved theoretical problems. In that way, they contributed to ‘experimental mathematics’. Among others, they proposed computer aided methods of solving linear functional equations, and (together with R. Quintero) presented an animation related to a convex-like property ([35] [36]).

1.2 Quantification of Artificial Mathematical Capabilities

While developing the new notion of mathability, a need for introducing a quantification of artificial mathematical capabilities appeared (among others: [5] [18]). A simple calculator, a graphical calculator, a computer algebra system (operating with formal mathematical language) – require different skills from a person using the devices. Designating a particular measure of demanded mathematical abilities could simplify answers to some important questions, for instance such as:

- What sort of a smart device is allowed to be applied during an examination?
- Who and in which form should control whether the device does not exceed the admissible capabilities?
- What mathematical skills are required from a candidate for a given position?

Until now, no formal scale of quantification has been introduced. Nevertheless, we can say that a simple calculator has a low mathability level, while a graphical one has a higher level. Moreover, we can designate a high mathability level to devices on which it is possible to install any mathematical application. Thus, the higher mathability level of the application, the higher level of the device. In this sense any smartphone is of a high mathability level, since we can easily use an advanced application solving computational and formal science problems, for example in mathematics, physics, chemistry, statistics, etc.

As an example we can point out such applications as Photomath or Wolfram Alpha, which are popular among students of schools and universities. What is interesting, the two high mathability level applications can be successfully applied by users with a basic level of knowledge and understanding of the mentioned sciences. The applications are frequently overused for finding solutions instead of solving problems by the users on their own. It leads easily to unrecognized errors, making mistakes in concluding or building mental maps. Having the method of quantification of mathability it could be possible to warn users they are expected to operate on the appropriate level. Unfortunately, easy access to advanced, professional applications and devices causes that they are applied in learning by people who are not ready to understand results produced by the equipment. This is why the notion of mathability was naturally broadened to the educational aspect.

1.3 Educational Aspect of Mathability

Technological revolution brought various devices supporting problem solving such as, for instance, statistical inferences, complex calculations, deriving formulas. The devices provide also an aid for education, research and personal development. Taking into consideration young people's habits new teaching methods were developed and introduced. The aspects were investigated, among others, in [10] [19-22] [32] [44-47]. The authors proved meaningful changes in perceptive templates of the young generation. On the other hand, several risks were described related to overusing smart technologies and internet sources of knowledge, applied with no reflection or when the knowledge is not sufficiently understood. An attention was also paid to the changing role of educators (see: [18] [22] [23]).

All above mentioned investigations show how the notion of mathability evolved and how its educational aspect was broadening. Starting with developing teaching and learning computer science with mathematical methods, we showed how to apply computers in mathematical education. Moreover, it was exemplified how to use computer assisted activating methods in teaching sciences and what constructive learning means in this context (read more in: [23] [30]). Finally, we discussed applicability of the methods for soft-skills training [18]. Currently we can talk not only about mathability in the sense of processing data (by programming or with human brain), building knowledge with formal algorithms, creating mental structure of knowledge but about ability of learning in a general sense, which may be called "Learnability".

The deep need for possessing Learnability is drawn in the following chapter.

2 Contemporary Labor Market

2.1 Cognitive Infocommunications and Skills Demanded by the Labor Market

The technical revolution causes dynamic changeability of habits, lifestyle, and labor market. It is difficult to predict what sort of jobs will be offered in the future to graduates who are beginning their school education nowadays. Moreover, any employees must be aware of the need to change their profession in the future. The problem is traditional schools do not keep up with the changes and they require an instant adaptation. The generation gap between conservative teachers and modern young people results in demotivation for learning with traditional methods. What kind of knowledge and abilities ought to be offered by schools may be explained by the requirements of employers.

By the reports *Responsive and responsible leadership* (Davos 2017, 2019, https://www.weforum.org/focus/responsive-and-responsible-leadership), we can call the contemporary labor market as **anytime, anywhere**. Quite often, a job can be performed at a time chosen by an employee in a place in which they currently stay. It is observed that "smart" technology facilitates variety of processes and there is no company running its business without computers. There are super-structures created in organizations and they build new eco-systems of media, where social media are the most important ones. The social life and personal soft abilities are as important in the real as in the virtual world.

This is why employers expect new sorts of their employees' competences. Among others, we can point out:

- Accepting and being ready for changes
- Quick self-development
- Creativity and creative problem solving
- Critical thinking

The competences open the list of the most demanded professional skills of the future labor market. A short survey done among employers (cooperatives of Institute of Mathematics of Kazimierz Wielki University) shows that currently the list may be completed with:

- Effective interpersonal communication, negotiation
- Goal-oriented time and tasks management
- Human resources management
- Building teams and team cooperation
- Self-assertion and emotional intelligence
- Concluding and making decisions

The competences are taught to students of management and related majors, but they are demanded from any employees and candidates. It creates a need for a quick gaining of new knowledge and skills, popularly called soft skills. They mean a combination of interpersonal (human) skills and personal (career) attributes (read more in: [50]). It should be mentioned that the employers clearly state that hard knowledge of facts, algorithms, connotations, and hard (work related) skills currently play smaller role than soft skills. However, schools still put emphasis on teaching rather hard competences. Hence, the training of soft skills must be organized by the employees on their own, with support of employers, simultaneously with developing professional career.

Taking the above into consideration we can claim that not remembering but:

- Immediate selection
- Evaluation
- Data processing
- ICT aided making decisions

will become crucial human abilities aiding a process of quick gaining of knowledge and being the foundation of cognitive flexibility and readiness for changes.

2.2 Computer-aided Preparing to Professional Life

Described in the previous chapter, interpersonal skills and career attributes (known as soft skills) are the most important competences demanded by employers but not sufficiently trained at schools and universities. For example, according to the Polish law, a university curriculum must contain social or humanistic subjects providing competences, achieving which takes at least 5 ECTS points (i.e. between 125 and 150 hours of an average student’s workload). It means that the majority of subjects are related to the professional knowledge considered by academics to be essential for becoming a specialist in the given branch of science.

The students’ point of view is completely different. The following survey was done among 41 students of computer science (1st year) and mathematics (3rd year). Students come from 11 distant countries and study at two different universities in Bydgoszcz, Poland. Majority of them (95.7%) know clearly what they want to achieve in their private and professional life. 57.7% of the students declare they choose the major in order to possess skills sufficient to get any good job in their future. It means they expect to learn general competences, not necessarily specific for their planned profession. In fact, the job they plan to perform is not necessarily related to IT or mathematics. All students claim they achieve competences for their future job in informal education, attending certified courses or some internet courses (not ending with any certificates). Hence, we can conclude that academic education is not sufficient to let students prepare for their professions.

The thesis is also proved by results of a survey done by McKinsey Global Institute (MGB) published by J. Manyika in *Technology, jobs, and the future of work* (for details, we refer to https://www.mckinsey.com/featured-insights/employment-and-growth/technology-jobs-and-the-future-of-work). About 40% of interviewed employers (coming from 9 countries) declare that their employees do not possess required skills to perform their job well enough. 60% of respondents complain that candidates are not ready to deal with their professional obligations. 49% of employers observed that the job is not a challenge for their employees. An independent research was done in Industrial and Technological Park in Bydgoszcz, Poland. The answers given by employees clearly show that 70% of them do not have sufficient skills to do their daily routines. On the other hand, they feel their talent and personal competences are underutilized. It may mean they are employed on an improper position. This is why about 49% of them state that their job is not a challenge. Comparative results are presented in Figure 1. Blue bars refer to the international survey and the red ones – to the Polish research.

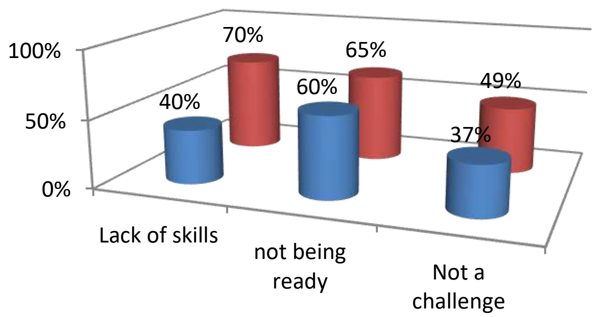


Figure 1
Employers and employee's opinion on readiness to perform a job

Now, it is worth investigating how young people get ready to meet the requirements of their current or future employers. From the survey done among students (mentioned above) we know that about 50% of them search in the internet for knowledge and competences deepening their interests but not related to the academic requirements, more than 38% of the examined students use internet to gain knowledge and skills deepening professional competences related to their current or future work. They usually use Google and YouTube to find professional advice. If they need a book they prefer pdf files (58.4%) and rarely borrow books from a library (15.3%). When browsing webpages, they frequently use positions from the very top of the list given by the browser (about 50%) and only 23% declare they read webpages advised by an authority. Taking into account that employees must deepen knowledge and skills in the process of a fast and adaptive education and self-education, the conclusion leads to the requirement of careful and deepened training of responsible ICT application in order to use internet and smart devices for self-improvement and development.

We should notice also that:

- Students finish or break their education at the bachelor level (for example, at Kazimierz Wielki University, on average, out of more than 90 graduates of computer science only 15 continue their studies)
- Some competences are taught by employers directly or indirectly by organizing proper courses and trainings
- The employees must learn while doing their duties

Concluding, future employees will continuously use multimedia devices to acquire knowledge, abilities, and even to build new relations. The dispersed and remote learning seems to be inevitable.

3 Learnability - Mathability for Education

Changeability and impossibility of predicting the future of the labor market offer as well as the requirements of a fast adaptive self-development of employees oblige the education system to prepare pupils and students to the responsible, conscious and useful life-long learning. It would be valuable to transform the system from traditional teaching to the computer aided constructive learning.

3.1 Cognitive Portrait of the Young Generation

A simple example presented in [20] (for further explanation, we refer to [41]) shows that the process of collecting and processing data is different for people trained in reading traditional (printed) texts and those who browse hypertexts.

Teenagers and university students are very good at essential reading of non-linear texts of webpages, searching for keywords and immediate matching, comparing and concluding. Unfortunately, solving problems or searching for new knowledge they are satisfied with sketchy solutions, they do not pay enough attention to deep understanding. They minimally reflect on the obtained result. Moreover, they use the easiest accessible sources of data. The most popular ones are Wikipedia and YouTube (more than 95% of respondents of the research mentioned above), and webpages chosen from the top of a browsing list (50% of respondent). 38% of the examined students relay on advised, specialist webpages. Among them professional courses on www.udemy.com are one of the most popular. Learners have a choice of multiple topics, information and methods. They limit their interests, knowledge and abilities to the preferable ones, not necessarily building complex mental structures.

All the above proves young people’s habit of superficiality and rapidly decreasing interest in formal education. For instance, our long term observation of students’ (aged 13-19) attitude towards learning mathematics showed declining interest in the subject, what is observed also at Polish universities (the number of candidates significantly decreases every year). Informal education plays more and more important role, and diplomas of higher education institutions lose their value. This is why the ability of efficient learning, Learnability, becomes more and more significant.

3.2 Foundation of Learnability

As we discussed above, as far as the education is concerned, teenagers and students are critical and lose their motivation for formalized forms of studies. Frequently, they do not learn what they are enforced to learn by the system of education, but what they need in order to build their consistent career plan.

Learners acquire knowledge and competences on the informal basis hence they learn what they consider important for them, and what, in their opinion, makes sense. Forced to learn what they find uninteresting or useless, they choose knowledge partially, incoherently; hence, the material is easily forgotten. Having a variety of offers on-line, they choose issues toward which they possess deep, inner motivation. They understand education as an investment into their future.

The described features are well known elements of learning patterns typical for adults (for details, we refer to [42] [43]). However, nowadays they are clearly observed already among teenagers. They demonstrate their interest in a few subjects and ignore other ones (bearing the consequences with bad marks).

What is a systemic answer to the changing requirements of pupils and students? Some solutions have already been proposed. For example, students of high schools are given a collection of subjects they can choose to learn on an advanced level (moreover, for instance in Polish high schools, not all subjects are obligatory). Projects and other activating methods are implemented more frequently to create an opportunity for learning a team cooperation, collecting data and applying knowledge, etc. Methods of selection, evaluation and ordering information are considered to be key competences and included in the school curriculum descriptions. Moreover, they are advised to be introduced already on the preliminary stage of a cognitive process. To obtain such goals we do need activating methods, engaging students, intensifying their motivation.

The reality seems to be far from those demands. Teachers declare knowledge of variety of activating methods. However, they apply them rarely since they are time consuming and do not engage the whole group of students (results of surveying educators, participants of workshops *Coaching in education,* conducted by K. Chmielewska during: nationwide conference *First of All Human. Bridges Instead of Walls*, Bydgoszcz, Poland 2019). The school curriculum consists of issues enabling students to collect (in the final test) a number of points which is sufficient to enter a school of the next level. The knowledge of definitions, facts, algorithms is still more important than soft and key skills mentioned above.

Additionally, among the known activating methods the educators did not mention *Problem Based Learning (*PBL) or *Inquiry Based Science Education* (IBSE) which fulfil all the requirements of modern constructive education.

Majority of the examined teachers declared also training of selection, evaluation, and ordering information. However, they were not able to point out such moments during their classes, when they regularly train these competences.

The conclusion seems to be obvious. The school and academic system needs to be transformed. It was also stressed in reports from the World Economic Forum Annual Meeting Davos 2020 (https://www.weforum.org/events/world-economic-forum-annual-meeting-2020/themes/society-future-of-work).

3.3 Constructive Learning as a Way for Achieving Learnability

To meet both requirements of the future labor market and demands of learners we propose a constructive learning method (which was described with details in [20]).

By constructive education we mean, briefly speaking, building knowledge and competences by experience. The idea precursors are J. Dewey ([31]), Gy. Pólya ([49]), J. Bruner ([14]) and J. Piaget ([48]). According to Dewey, school education should present real life problems and students should be given a chance to experiment.

Educational experiments (results of which are presented in [17-20]), and further investigation (discussed, among others, in [42] [43-47]) allowed us to observe a common cognitive pattern which can be explored as a method of conscious and responsible learning. In this method, assimilation of knowledge necessary for solving the given problem is represented by:

- Searching for the knowledge – a definition or broader description of the problem, examples illustrating the notions.
- Understanding notions, methods and examples.
- Following the way of the analyzed solutions.
- Finding an own solution of the original problem.
- Reflection and assessing the obtained result and applied methods.

Let us compare the method with Bloom’s and Kolb’s theories [9] [42] [43]). The method originating from the 50s and 70s, can be considered from a new point of view, taking into account the fact that young people learn like adults and, using smart devices and distant sources of knowledge, they construct their knowledge.

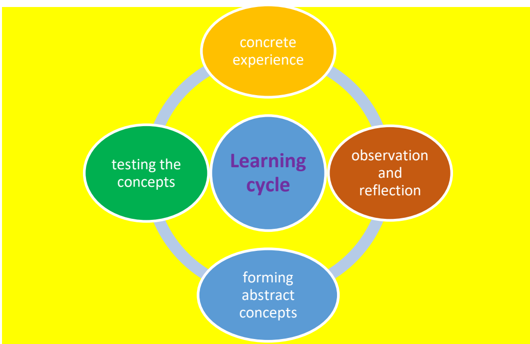


Figure 2

Kolb’s experiential learning cycle

Students browse knowledge sources, search for information and examples

The step stands for Bloom's *stage of remembering*. In Kolb's cycle we could compare it to the stage of a *concrete experience*.

Since information is easily accessible, young people do not feel the need to memorize it. Instead of remembering, they find necessary information as often as they need it. It seems to be a strong habit of youth. Therefore, it should be trained to become a powerful tool of their self-improvement. For that, the next step is even more important.

Students evaluate and select appropriate information fitting the prior knowledge system

Two Bloom's stages are consistent with this step: *analyzing* and *comprehending*. It also corresponds to Kolb's *observation and reflection stage*.

It is extremely important to choose information which is understandable and credible. Skipping the step brings undesirable effects, for instance, building a false foundation for further investigation. Since it is a way of discovering, it is effective and has a great impact on durability of the possessed knowledge. Any mistake made that way will be hard to eliminate. It explains the fact why the skills of selection and evaluation must be trained at school even on the elementary level, since children absorb information collected from several sources, among which internet is the most common.

Students assimilate the new knowledge into the prior knowledge system

The step corresponds to Bloom's *applying* and Kolb's *testing the concept*.

Students compare, build analogies, search for relations, and conclude. They can try to repeat the new knowledge with other examples of usage. For instance, if they learned an algorithm they try their own computation with other input data. The new knowledge or skill becomes an applicable tool.

Students interpret new knowledge, which is adequate to Bloom's *synthesizing*

They have just gained new knowledge or experience. They search for further examples of usage. It is valuable to ask questions such as *what if*:

- We change one condition,
- We apply it for other input data.

Students join pieces of information or part of a method together in order to create new meaning, structure or algorithm.

Students reflect on an overall result, evaluate new knowledge or methods

It corresponds to Bloom's *evaluating* stage.

The last two steps are adequate to Kolb's *forming abstract concepts* stage. From that point the cycle starts again from the beginning since new questions should arise and make students search for more information and new methods. Students try to find out a new use of the result, broaden its usefulness, find its limitations, etc.

The described method of constructive learning goes beyond typical computer aided education, understood as a usage of internet sources of knowledge or applying popular educational applications (e.g., for Mathematics, Cabri, and Geogebra at schools, or Wolfram Mathematica, Statistica, MathLab, Mapple at universities). This is an applicable method of discovering, building definition, finding relations, recognizing properties, based, among others, on pupils' habit of using smart devices.

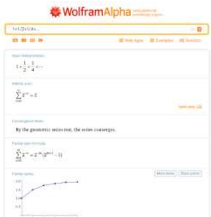


Figure 3
Tools aided constructive learning of Science

4 Applicability of Mathability Methods

Describing educational aspects of mathability we examined some activating methods and elements of constructive learning (see: [16]-[23] and [30] [38] [43]). Here we would like to discuss how the methods can be applied to general education, also for soft skills training.

4.1 Common Factors of Activating Methods

As Benjamin Franklin said: *Tell me and I forget, teach me and I may remember, involve me and I learn*, methods directly involving students into the process of education are much more effective and lasting.

It is worth mentioning some common factors of activating methods.

A change taking place in the students

Students, discovering knowledge and learning by experience, become more confident. They understand their right to make mistakes, not knowing is considered to be a natural state which provokes investigation. Doubts are reasons for asking questions and questions are natural tools of examiners. Instead of repeating: *I do not know since I am too weak*, they say *I do not know, so I will learn it. Let me try.*

Stable effects

As it was mentioned above, discovering and experiencing has a strong impact on learners thanks to the great effectiveness of the methods. Students memorize the material faster, remember what they have learned for a longer time, and understand it deeper.

Applicability for variety of problem solving

Among the steps of constructive learning there is broadening of usefulness of the created result. As a result, we mean not only what students obtained after a chemical experiment or mathematical computation but a way in which they gained their experience, as well. Hence, a method they have just applied for patient, diligent, dogged finding solutions may be applied for solving their life problems, such as planning their future, managing tasks, team co-operation, etc.

We may ask what the linking element is. The answer appears easy: **unconstrained engagement following deep and inner motivation, natural interest, and need of self-improvement**.

The methods described below are based precisely on the assumption that learners are deeply and internally motivated to obtain their goals, they take part in the process of education following their true interest and necessity.

4.2 Edu-Coaching

Coaching is a personalized and holistic approach, taking into account the fact that everyone is different and that our lives and development take place simultaneously in many areas. It is based on the relationship between coach and those coached.

Educational coaching (following a definition presented at www.coachingedukacyjny.edu.pl/#czym-jest-coaching) is an educational process aimed at developing competences of educators and achieving better learning outcomes by understanding children's behavior, skillful work with them and effective motivation to learn.

On the other hand, that is a process focused on improving and strengthening learning and development by increasing self-awareness and personal responsibility (for more details and examples we refer to [23] [37] [38]). A coach accompanies a student by asking questions, listening actively and challenging the student in a supportive and motivating way.

By J. Rogers ([51]) there are six fundamental conditions under which coaching must be conducted. We will explain how the conditions are fulfilled in education.

1) **A student owns necessary resources**

   Frequently students in a problematic situation resign from finding a solution, since they think the problem is too difficult or they are too weak to solve it.

The same students, motivated appropriately, reveal knowledge and competences sufficient to break their impasse, name the problem and find a solution which they can carry out on their own. Working with their coach they discover notions, facts, algorithm necessary to obtain their goals. A student's resource can also be the awareness of the right to lack knowledge and considering having a problem as a natural state.

2) **A teacher (coach) asks questions so that the student begins to use their resources**

   By asking questions the teacher (coach) creates a space for the student to think out all necessary answers. The questions do not suggest any answer. They only encourage and inspire to examine the problem from a different point of view or in relation to the student's previous experience. They may lead to a conclusion what is the missing part of knowledge, completing which will guarantee a success.

3) **Coaching refers to the past, present and future of the coachee**

   The past in the context of education means the prior knowledge and experience of the student, his previous habits and talents. The present refers to the current abilities and possibilities helpful for obtaining the desired objective. The coach accompanies the student while analyzing the sources. Moreover, the coach helps to reveal what is a real obstacle in breaking the problem. It enables the student to start working on his motivation, patience, creativity, and other personal abilities.

4) **The topic is chosen by the student**

   The student explains the teacher what the topic of the current work is, what the challenge is. The coach cannot insist on choosing another subject. Frequently, the student finds out the reason of the problem and decides to change the topic, but every time it is the decision of the student. The topic may be either a subject-oriented task or a personal ability which the student wants to improve.

5) **The student and the teacher are partners**

   During the coaching process the student and the teacher play role of partners. This is what distinguishes the method from mentoring and tutoring, where the teacher is a master, more experienced colleague or an expert.

6) **The goal of the cooperation is solving a problem or a change**

   Depending on the topic chosen by the student, the couple coach/coachee works on solving the problem or the student's personal **change**.

Solving problems or breaking impasse, in fact is a sort of defining a challenge which motivates the student for deepened engagement. In this case we talk about **transactive coaching.** It is a perfect way to support constructive learning in which the student, answering questions, builds knowledge and gains skills. It is also a kind of support for learning through ICT aided discovering. The coach will take care to make the student avoid risks related to superficiality, lack of reflection and others we described above (more about risk of un-guided self-education one can find in [17] and [20]).

The change is related to obtaining and strengthening personal abilities. This is the case of **transformative coaching**. It is significant for strengthening motivation, discovering one's own capabilities, advantages and skills, abandoning limiting beliefs, revealing or improving talents, awaking self-confidence, etc.

Apart from the above application, coaching can be useful to improve interpersonal communication between:

- The teacher and the student
- The teacher and the student's parents

There are only two basic tools necessary to apply coaching in education. One of them is asking open and non-suggestive questions. It demands giving the student time to response. Even if the student is silent, the coach should not interrupt or answer instead of the coachee. Simultaneously, the student should feel free to take the time and not to answer immediately. The situation requires a deep trust and strong, partnership relation between the coach and coachee.

The other tool is known as, active listening. It means listening to words, emotions, attitudes, and behaviors, which requires the coach to focus totally on the student. Questions asked by the coach must follow what was heard from the student. The difficulty of the meeting consists in the fact that it cannot be planned and the coach must react on the spot to each student's message.

Since coaching is strictly formalized (e.g. the coach should not suggest any solution, nor give advice) other methods are more popular and more frequently used at schools. However, coaching in its pure form, is the most effective method of personal development thanks to the fact that the total responsibility belongs to the coachee. Other methods, for example tutoring and mentoring, share responsibility between the student and the teacher. Thanks to possibility of giving advice they might seem to be faster, i.e. the student achieves goals earlier or with less workload, but the effect may be less lasting.

Interesting and valuable results on Edu-coaching methods in 3D VR environments can be found in [39].

4.3 Tutoring

Tutoring, similar to coaching, is a holistic approach based on the relationship between a tutor with a tutee. The tutor is a facilitator of the development process the goal of which is **discovering talents and developing the skills of independent thinking and creation.**

The method is based on the principle of “Personalism” where a person is treated with awareness that they have their values, past experiences and plans and they need to be taken into account when conducting training. The tutor, as an expert in their branch, leads their tutee to the success asking inspiring and provoking questions. If necessary, the tutor can advise or suggest, can propose literature or directions of development. After each meeting the tutee is given a topic of an essay to write, in which they present their thoughts and justification of their personal ideas.

Again, we can distinguish two kinds of methods: focused on discovering and developing talents (developmental tutoring) or focused on developing knowledge in a chosen area (scientific tutoring). In the second case, the tutor must be an expert in the area of training.

The basic tools and methods applied in coaching are useful for tutoring, as well. The master asks open questions following the tutee argumentation, their needs and problems. The tutor listens actively to the tutee, reacting immediately to their talk, emotions, and behavior, challenges the tutee and proposes steps of their development.

Both coaching and tutoring are applied to the best students, who are ready to accept what they may reveal during the process. The methods are awards for the students’ prior achievement. The methods described below are appropriate for any students and pupils.

4.4 Soft Skills Training

As we wrote in chapter 2 (following [50]), soft skills mean a combination of interpersonal (human) skills and personal (career) attributes. From own experience and interviews with some coaches and business trainers we can claim that among the most popular skills chosen by clients to be trained the following ones can be listed:

- Team cooperation
- Team building
- Team management
- Time and task management
- Emotional intelligence
- Negotiation
- Communication
- Assertiveness
- Personal and team effectiveness (which are composed of some of the above skills on this list)

The above skills coincide with all of the needs of the labor market and demands of students completing their education before starting their jobs (as described in Chapter 2).

A trainer of soft skills uses coaching methods of asking inspiring and provoking question, but at the same time presents the theory necessary for understanding processes of management, communications, etc. A great and difficult ability of the trainer is to build a training on the spot, following a situation created at the given moment or responding to particular trainee's needs. The trainer must be able to manage emotions of the trainee or of the whole group of participants.

The described method is called a "Consultment" training. It is more efficient than any course presenting methods and giving instructions in a stiff way, where examples are prepared in advance without examining participants expectation. During consultment training, the examples and exercises are being composed at once, according to the current need of participants. The method is interactive and follows the action of the group and each individual participant.

Like previously described methods, soft skills training is a holistic, but group process, which means that not only a single trainee but the whole group will evolve and change during the process. Being aware of the stages of the group process, an educator may manage the development of the class as a group or a students' group as a team, simultaneously with teaching them the subject material.

4.5 Networking

By networking we mean the exchange of information and ideas among people with a common profession or special interest, usually in an informal social setting. Objectives of networking include, among others, expanding circles of acquaintances of participants, finding out ideas about job opportunities in their fields, and increasing their awareness of news and trends in their fields or in the greater world.

It is performed, for instance, as trade shows, seminars, and conferences, which are designed to attract a large crowd of like-minded individuals (more details may be found at www.investopedia.com).

Let us present a Polish example of after school extracurricular education, which includes also networking as a method of exchanging ideas (details may be found at www.kopernik.org.pl/en/projekty-specjalne/).

The project is coordinated by Copernicus Science Centre in Warsaw, Poland. Up till now, it has involved more than 500 Polish schools and universities into an active cooperation. Kazimierz Wielki University in Bydgoszcz as a partner in the project will examine the effectiveness of the constructive education.

Pupils and students are inspired to observe, experiment, ask questions, and seek answers in biology, physics, chemistry, mathematics, and other sciences. The applied methods change the common teaching to the system of constructive learning. Unfortunately, since students choose the topic of each meeting it is hard to predict objectives which will be achieved. Because of that the method cannot be applied on the regular basis, since the formal education system requires to plan in advance a list of learning outcomes. That is not possible in the project. The only objectives planned to be achieved are soft skills such as, for instance, asking questions, effective communication, team cooperation, self-motivation, adequate self-esteem, deepening self-confidence, etc. What students will learn depends on their interest, but they definitely will obtain personal abilities useful in their life and professional career.

Conclusions

Perception and learning practices are strongly influenced by the habits of using multimedia, smart devices and the tools of cognitive Infocommunication. The system of education needs to be transformed, since modern education should take advantage of new habits, in order to help students to be ready for life-long learning and to find a satisfying position in a dynamically changing labor market. Teachers, in the new formal system, should play the role of facilitators, mentors or tutors to assist pupils and students, in gathering information, making conclusions, discovering, and solving problems.

Students may apply high level Mathability devices and applications, as well as, use multimedia knowledge sources, however, they must be guided by the facilitators in order to avoid risks of superficiality and building inadequate knowledge structures.

Acknowledgement

This work was supported by Kazimierz Wielki University in Bydgoszcz.

References

[1] Baranyi P., Csapó Á.: Definition and synergies of cognitive infocommunications, Acta Polytechnica Hungarica, Vol. 9, pp. 67-83, 2012

[2] Baranyi P., Csapó Á.: Revising the concept of generation CE-generation of cognitive entities, in 6th IEEE Conference on Cognitive Infocommunications (CogInfoCom) IEEE 2015, pp. 583-586

[3] Baranyi P., Csapó Á., Sallai Gy.: Cognitive Infocommunications (CogInfoCom) Springer, 2015

[4] Baranyi P., Csapó Á. And P. Varlaki: An Overview of research trends in coginfocom, in 18th International Conference on Intelligent Engineering Systems (INES) IEEE, 2014, pp. 181-186

[5] Baranyi P., Gilányi A.: Mathability: emulating and enhancing human mathematical capabilities, 4th IEEE Conference on Cognitive Infocommunications (CogInfoCom) IEEE, 2013, 555-558

[6] Biró P., Csernoch M.: Deep and surface metacognitive processes in non-traditional programming tasks, 5th IEEE Conference on Cognitive Infocommunications (CogInfoCom) IEEE, 2014, 49-54

[7] Biró P., Csernoch M.: The mathability of computer problem solving approaches, 6th IEEE Conference on Cognitive Infocommunications (CogInfoCom) IEEE, 2015, 111-114

[8] Biró P., Csernoch M.: The mathability of spreadsheet tools, 6th IEEE Conference on Cognitive Infocommunications (CogInfoCom) IEEE, 2015

[9] Bloom B. S.: Taxonomy of Educational Objectives: The Classification of Educational Goals, Susan Fauer Company, Inc., 1956

[10] L. Bognar, F. Francsikne, P. Horvath, A. Joos, B. Nagy and G. Strauber, Improve learning environment for calculus courses, Journal of Applied Technical and Educational Sciences, Vol. 8, No. 4, 2018, pp. 35-46

[11] Borus G. Gy., Gilányi A.: On a computer programme for solving systems of functional equations, 4th IEEE Conference on Cognitive Infocommunications (CogInfoCom) IEEE, 2013, p. 939

[12] Borus G. Gy., Gilányi A.: Solving systems of linear functional equations with computer, 4th IEEE Conference on Cognitive Infocommunications (CogInfoCom) IEEE, 2013, 559-562

[13] Borus G. Gy., Gilányi A.: Computer assisted solution of system of two variable linear functional equations, Aequationes mathematicae, 2020, 94(4), 723-736

[14] Bruner J. S., Haste H.: Making Sense. The Child's Construction of the World, Methuen, New York, 1987

[15] Bubnó K., Takács L., The mathability of word problems as initial computer programming exercises, in: 8th IEEE Conference on Cognitive Infocommunications (CogInfoCom) IEEE, 2017, pp. 39-44

[16] Bubnó, K. T., Takács, V., Ambrus, A., & Vásárhelyi, É., Solving word problems by computer programming. In: Problem Solving in Mathematics Education, Eötvös Loránd University, Faculty of Science, Institute of Mathematics, 2014 (pp. 193-208)

[17] Bubnó, K., & Takács, V. L. Mathematical problem solving with Blockly

[18] Chmielewska K.: From Mathability to Learnability, in 10th IEEE Conference on Cognitive Infocommunications CogInfoCom) IEEE, 2019, 513-516

[19] Chmielewska K., Gilányi A.: Mathability and computer aided mathematical education, in 6th IEEE Conference on Cognitive Infocommunications (CogInfoCom) IEEE, 2015, 473-477

[20] Chmielewska K., Gilányi A.: Educational context of mathability, Acta Polytechnica Hungarica, Vol. 15, 2018, pp. 223-237

[21] Chmielewska K., Gilányi A.: Computer assisted activating methods in education, in 10th IEEE Conference on Cognitive Infocommunications CogInfoCom) IEEE, 2019, 241-246

[22] Chmielewska K., Gilányi A., Łukasiewicz A.: Mathability and mathematical cognition, in 7th IEEE Conference on Cognitive Infocommunications (CogInfoCom) IEEE, 2016, 245-250

[23] Chmielewska K., Matuszak D.: Mathability and coaching, in 8th IEEE Conference on Cognitive Infocommunications (CogInfoCom) IEEE, 2017, 427-431

[24] Csernoch, M., Dani, E.: Data-structure validator: An application of the HY-DE model, 8th IEEE Conference on Cognitive Infocommunications (CogInfoCom) IEEE, 2017, 197-202

[25] Csapó G., Sprego virtual collaboration space, in 8th IEEE Conference on Cognitive Infocommunications (CogInfoCom) IEEE, 2017, 137-142

[26] Csapó G., Placing event-action-based visual programming in the process of computer science education, Acta Polytechnica Hungarica, 2019

[27] Csernoch M., Miró P., Introduction to classroom sprego, Acta Didactica Napocensia, Vol. 9, 2016, pp. 1-14

[28] Csernoch M., Miró P. First year students' attitude to computer problem solving, in 8th IEEE Conference on Cognitive Infocommunications (CogInfoCom) IEEE, 2017, 225-230

[29] Demeter R., Kovari A., Katona J., Heldal I., Costescu C., Rosan A., Hathazi A. and Thill S., Effects of simulation software on student's creativity and outcomes, in 10th IEEE Conference on Cognitive Infocommunications CogInfoCom) IEEE, 2019

[30] Dergham M., Gilányi A., Application of virtual reality in kinematics education, in 10th IEEE Conference on Cognitive Infocommunications CogInfoCom) IEEE, 2019

[31] Dewey J.: Experience and Education, NY Kappa Delta Pi, New York, 1938

[32] Geszten D., Komlódi A., Hercegfi K., Hámornik B., Young A., Köles M., and Lutters W. G., A content-analysis approach for exploring usability problems in a collaborative virtual environment, in 7th IEEE Conference on Cognitive Infocommunications CogInfoCom) IEEE, 2016, pp. 67-88

[33] Gilányi A.: Characterization of Monomial Functions and Solution of Functional Equations Using Computers (Charakterisierung von monomialen Funktionen und Lösung von Funktionalgleichungen mit Computern, German) PhD Thesis, University of Karlsruhe, 1995

[34] Gilányi A.: Solving linear functional equations with computer, Math. Pannon., 9:57-70, 1998

[35] Gilányi A., Merentes N., Quintero R.: Mathability and an animation related to a convex-like property, 7th IEEE Conference on Cognitive Infocommunications (CogInfoCom) IEEE, 2016, pp. 227-231

[36] Gilányi A., Merentes N., Quintero R., Presentation of an animation of the m-convex hull of sets, in: 7th IEEE Conference on Cognitive Infocommunications (CogInfoCom) IEEE, 2016, 307-308

[37] I. Horvath, "Digital life gap between students and lecturers," in ´ 2016 7th IEEE International Conference on Cognitive Infocommunications (CogInfoCom) IEEE, 2016, pp. 353-358

[38] I. Horvath, "The IT device demand of the edu-coaching method in ´ the higher education of engineering," in 2017 8th IEEE International Conference on Cognitive Infocommunications (CogInfoCom) IEEE, 2017, pp. 379-384

[39] I. Horvath, "The edu-coaching method in the service of efficient teaching of disruptive technologies" in Springer book Cognitive Infocommunications, Theory and Applications, 349-363

[40] Katona J., Kovari A., Examining the learning efficiency by a brain-computer interface system, Acta Polytechnica Hungarica, Vol. 15, 2018, pp. 251-280

[41] Kirschner P. A., De Bruyckere P.: The myths of the digital native and the multitasker, Teaching and Teacher Education 67:135-142, 2017

[42] Kolb D. A.: The Learning Style Inventory: Technical Manual, McBer, Boston, 1976

[43] Kolb D. A., Fry, R.: Toward an applied theory of experiential learning, in C. Cooper (ed.) Theories of Group Process, John Wiley, London, 1975

[44] Kovari A., CogInfoCom supported education: A review of CogInfoCom based conference papers, in: 9th IEEE Conference on Cognitive Infocommunications (CogInfoCom) IEEE, 2018, 205-224

[45] Kovecses-Gosi V., Cooperative learning in VR environment, Acta Polytechnica Hungarica, Vol. 15, 2018, pp. 205-224

[46] Petkovics I., Digital transformation in higher education, Journal of Applied Technical and Educational Sciences, Vol. 8, No. 4, 2018, pp. 77-89

[47] Rigoczki C., Damsa A., and Gyorgyi-Ambro K., Gamificationon the edge of educational sciences and pedagogical methodologies, Journal of Applied Technical and Educational Sciences, Vol. 7, No. 4, 2017, pp. 74-88

[48] Piaget J.: Studies in Child Psychology (Studia z psychologii dziecka, Polish), PWN, Warsaw, 1966

[49] Pólya Gy.: How to Solve It, Princeton University Press, Princeton, 1945

[50] Robles M., Executive Perceptions of the Top 10 Soft Skills Needed in Today’s Workplace, Archived 2016-08-12 at the Wayback Machine, Business Communication Quarterly, 75(4) 453-465

[51] Rogers J., Coaching Skills. A Handbook, Open University Press UK Ltd., 2008

[52] Szi B., Csapo A., An outline of some human factors contributing to mathability research, 5th IEEE Conference on Cognitive Infocommunications (CogInfoCom) IEEE 2014, 583-586

[53] Török M., Tóth M. J., and Szöllősi A., Foundations and perspectives of mathability in relation to the coginfocom domain, in 4th IEEE Conference on Cognitive Infocommunications (CogInfoCom) IEEE, 2013, 869-872