

Study of Algorithmic Problem-Solving and Executive Function

Attila Kovari

Department of Natural Sciences, Institute of Engineering, University of Dunaújváros, Táncsics M. út 1/A, H-2400 Dunaújváros, Hungary
kovari@uniduna.hu

Abstract: Many processes make up cognitive-communication, such as orientation, attention, memory, problem-solving and executive function. Executive functions are necessary for the cognitive control of behavior and refer to some basic cognitive processes such as attentional control, working memory and cognitive flexibility. In this paper, a study is presented, which examines the relationship between algorithmic problem-solving and executive function. Executive functions are examined using Hanoi-tower test and algorithmic problem-solving is investigated using a flowchart-based debugging test. The results indicate that a relationship can be found between problem-solving requiring algorithmic thinking and the executive function. The results received may have a positive impact on the efficiency of education of programming through the dedicated development of executive functions.

Keywords: executive functions; problem-solving; programming; Hanoi-tower test

1 Introduction

Many processes make up cognitive-communication, such as orientation, attention, memory, problem solving, and executive function. Executive functions are a set of cognitive processes that are necessary for the cognitive control of behavior, a higher-level of cognitive skills to control and coordinate other cognitive abilities [1]. These functions refer to some basic cognitive processes, such as, attentional control, cognitive inhibition, working memory and cognitive flexibility.

The executive function has high significance from the aspect of human behavior organization, however, there is no uniform definition on it [2], nor any agreement what exact partial operations make up executive functions. According to [3], we consider executive function as a set of abilities, which enables us to represent a target to be achieved, to establish a behavior plan to achieve such target so, that meanwhile, we are monitoring both the environment and our actions, and if necessary, flexibly adjusting the plan worked-out to achieve the particular target.

However, not every behavior requires control implemented by executive functions, such as reflex (e.g. muscle contraction upon pain stimuli) in very simple, routine actions (e.g. touching a glass on the table).

executive functions are always necessary if the target cannot be achieved by direct action. It is important, however, that this behavior control is implemented through the complex control of cognitive and other psychological processes (emotions, passions) being the basis the behavior [3]. Executive functions play a major role in all complex cognitive tasks, such as problem-solving, working memory function, lingual processing and communication. Executive function has therefore principal significance from the aspect of organizing adaptive, problem-solving human behaviors and psychological processes. The disorder of different components of executive functions may seriously limit both learning and adaptive abilities [4].

The behavior process described thus also fits well to solving a problem designated as a target, where the steps of the solution or the evaluation of the steps of the solution in the problem era requires the continuous re-thinking of solution options. During computer software development, the determination of steps leading to the solution of the task designated as the target, then it's coding into a series of instructions workable for the computer are the result of a similar process. In the analysis of this field, modern methods provided by cognitive info-communications [5][6] may be well utilized, that enable the examination of several aspects of cognitive processes [7]-[10].

There is a strong correlation between executive function and problem-solving controlled by conscious processes. The purpose of this paper is to create an analysis aimed at executive function and algorithmic thinking among students learning programming, which may provide grounds to determine further research directions.

2 Test of Executive Function

The measuring of executive functions is a quite complex methodology problem [3]. It makes another difficulty, that there is no clear boundary between routine, known and new tasks. The more routine-like is the actual task, the fewer executive functions are necessary for implementation [3]. The routine-novelty is, therefore, a very important dimension of measuring executive functions.

2.1 Wisconsin Card Sorting Test (WCST)

In the often used Wisconsin card sorting test, cards indicating simple figures must be grouped so (Figure 1), that the actual aspect of grouping must be figured out by the examination manager using the 'correct/incorrect' feedbacks on the testing subject's attempts [11]. For instance, the cards must be grouped based on the

shape, color or number of the figures presented on them. The examination manager, however, varies the aspects of grouping in fixed periods, without notifying about it, and this time, the testing subject has to give up his strategy although correctly figured out earlier, now being incorrect as soon as possible, and find the applicable new aspect. The test is primarily used to measure behavior/cognitive flexibility.

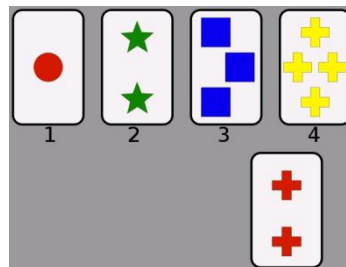


Figure 1

Wisconsin Card Sorting Test

2.2 Stroop Task

Stroop task processes are used to measure inhibition functions, thus, prepotent response inhibition. In classic Stroop-tasks, the subject has to answer which color was used when printing the particular color name [12]. The task is not trivial and considered as an executive function test, because – as we know from the failures – to correctly name the color of the ink, the response arising from the automatic reading of the word (as a prepotent response) must be inhibited when the color of the ink and the color indicated by the word do not match. Using the number/ratio of failures, the inhibition component of executive functions may be measured. The task has several variants.

2.2 Go/No-go Task

In this task type, a (usually simple) motoric response must be provided on every appearance of the warning stimulus, but the motoric response must be avoided when another warning stimulus appears [13]. Using the incidence of the two types of failures (motoric response when it should not be given; or the lack of response where it should be given), inhibition and flexibility aspects can be measured. Very versatile variants are used.

2.2 Tower of Hanoi and its Variants

Tower of Hanoi and its variants are procedures primarily used to measure planning, problem-solving [14]. In the classic Hanoi-tower task (Figure 2), there

are three pegs and three/five discs of various sizes, where the discs are strung on either peg, in sizes decreasing to the top. The task is to reproduce this shape on either another peg so, that only one disc may be moved at the same time, and bigger disc must not be placed onto the smaller one. This variant is not easy, several 'divergent', i.e. steps apparently to bring us farther from the solution must be executed systematically to achieve the goal. Using the number and ratio of correct/incorrect steps, as well as the time required to solve the task, planning and quick problem-solving function can be measured. Its simplified variant is the London-tower.

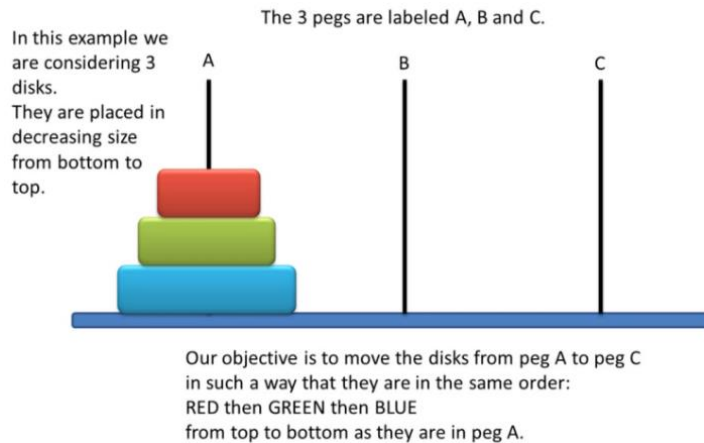


Figure 2

Tower of Hanoi using 3 discs

Tower of Hanoi is a mathematical game, where according to the game's rules, the discs must be relocated from the first peg to the last one so, that one disc may be relocated in each step, the bigger disc must not be placed onto smaller ones, and there are only three pegs available. The game was invented by Édouard Lucas, French mathematician in 1883 [15].

The solution of the puzzle requires concentration, quick planning, problem-recognizing & solving abilities, where the solution must be divided into appropriate steps, similar to a program algorithm.

1.1.1 The Mathematical Solution of the Tower of Hanoi

Let's make the lowest number of required steps: t_n . If there are $n+1$ discs, the bottom disc cannot be moved, as long as all discs above it are not yet relocated to the middle peg, which can be performed exactly in t_n steps. Then only the bottom (and biggest) disc may be moved onto the third peg, which may be performed in a single step. This is the t_{n+1} step. Then the n number of discs on the second peg may be relocated to the third peg, which can be achieved in additional t_n steps. Then the following correlation may be drawn up as to the number of steps:

$$t_{n+1} = t_n + 1 + t_n = 2t_n + 1 \quad (1)$$

$$\Delta t_n = t_n + 1 \quad (2)$$

As learned in the solution of first-grade linear inhomogeneous equations:

$$t_n = c \cdot 2^n - 1 \quad (3)$$

$$t_1 = 1; 1 = t_1 = c \cdot 2^1 - 1 = 2 \cdot c - 1 \rightarrow c = 1 \quad (4)$$

$$t_n = 2 \cdot n - 1 \quad (5)$$

1.1.2 PEBL Test of Tower of Hanoi

The Psychology Experiment Building Language (PEBL) is a free cross-platform system for designing and running computer-based experiments and tests. It is free to use, there are no licenses to maintain [16]. PEBL distributes a set of standard and novel test in the form of the Test Battery. The Tower of Hanoi is implemented in PEBL and uses a random problem generator (Figure 3).

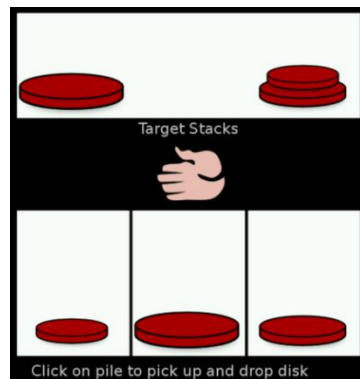


Figure 3

PEBL based Tower of Hanoi

Table 1

Summary test results of PEBL based Tower of Hanoi

sub	trial	size	shortest	startlab	endlab	startconf	endconf	success	steps	time[ms]
3	1	3	6	221	332	221	332	1	10	12456

To evaluate the individual differences, these parameters are important:

- The mean time to complete the task (time)
- The number of steps needed to complete the task (steps)
- The number of 'extra' steps needed (steps-shortest, where the fourth column is the shortest)

3 Executive Function and Algorithm Problem Solving

Programmers, following writing a program, can memorize only a small part of the program code, since human working memory is very limited [17]. Writing is already a very complex thinking process, and the same refers to writing a code [18]. However, coding must be always preceded by the creation of the algorithm leading towards the solution. This process is the algorithmic thinking, which is closely related to problem-solving since in both cases, we have to split up the route towards the solution of a problem into a series of subsequent steps. The conscious activity driven by the executive function affects the use of working memory. Due to the limited capacity of working memory, the programmer is unable to keep too complicated or code parts in their mind, as long as he interprets them as separate data. Once one learns how to organize these ideas and algorithms into samples, and create mental models on them, then one becomes able to solve more complicated tasks [19]. Executive function and problem-solving driven by conscious processes are therefore closely related to each other.

A preliminary study is introduced below, which aims to analyze the relationship between executive function and algorithmic problem-solving [20]. The executive function is examined using the Hanoi-tower test, whilst algorithmic problem-solving using debugging in an algorithm provided on a flowchart.

4 Study and Procedure

The evaluation of the relationship between executive function and algorithmic problem-solving is performed through the Hanoi-tower test, the successful debugging of the two errors in the algorithm described by a flowchart, and the comparison of mathematical and algorithmic abilities related to problem-solving.

The purpose of the test was primarily to determine how the interpretation of the algorithm according to the flowchart and finding the errors are related to the test results received regarding the Hanoi-tower task measuring the executive function. Also, how students' mathematical, algorithmic and foreign language skills are related to the test results of the Hanoi-tower task measuring the executive function.

In the test, 23 people, IT-Engineer students of 19-22 years, participated ($M=20.78\pm 1.28$). There were 11 females and 12 males among the testing subjects. Testing subjects were volunteers.

As a first step to prevent the prior knowledge play a role in the interpretation of the flowchart, a test analyzing the previous knowledge of the basic elements of the flowchart was performed. The test consisted of 9 questions, aimed at the flowchart basic elements and their functions. Two test questions are indicated in Figure 4. In

the case of those students, where the certain knowledge of the flowchart elements was inappropriate, the information related to the failed questions were discussed.

What is the name of this flowchart element?

What is the end of this algorithm?



start/stop
instruction
decision
process
document

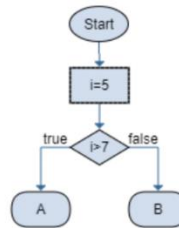


Figure 4

Sample test questions for flowcharts

In the second step of the investigation, the flowchart indicated in Figure 5 had to be observed by the students for 2 minutes. The students were aware that two errors were hidden in the flowchart, which they had to say when the time expired. The flowchart's elements were numbered, making the error easier to determine. In the flowchart, one of the two errors was the arrow marked with No. 23, which connects to an inappropriate place, and the other one was in the process marked with No. 13, where the $c[k]=a[i]$ index variants were mixed up.

Sort the elements with the T property of a series of N elements, where the T property means that the given number is greater than 5.

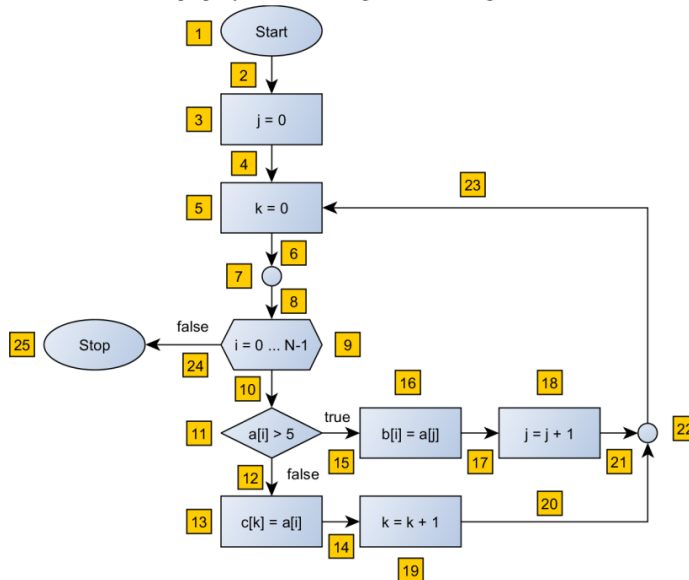


Figure 5

Flowchart test

In the third phase of the test, the students were practicing the Hanoi-tower test implemented into PEBL system three times, to learn what was the test about and how to execute it. Following the initial practicing, in the fourth phase of the test, they solved the Hanoi-tower task ten times in a row. The parameters, results related to the solution were saved by the PEBLS system into a file, which may be thus post-processed.

In the last phase of the investigation, the students answered an additional five questions using a questionnaire. The five questions were as follows:

1. How would You rate your mathematical skills? (1-5 on Likert scale)
2. How would You rate your algorithmic skills? (1-5 on Likert scale)
3. How would You rate your foreign language skills? (1-5 on Likert scale)
4. How long have You been dealing with programming?
5. How many types of programming languages have You learnt so far?

5 Results

The students' knowledge related to flowchart interpretation proved to be appropriate, some of them completed the test consisting of 9 questions flawlessly, whilst some with only 1-2 mistakes. The results are summarized in Table 2.

Table 2
Results of flowchart basic pre-knowledge test

Number of good answers (9 questions)	Number of test subjects
9	16
8	6
7	1

Following discussing the mistakes made regarding the questions, the test was continued with the second phase, the debugging on the flowchart. Table 3 and Figure 6 indicates how many of the testing subjects managed to find 0, 1 or both of the 2 errors on the flowchart.

Table 3
Results of flowchart debugging test

Number of errors found	Number of test subjects
0	5
1	13
2	5

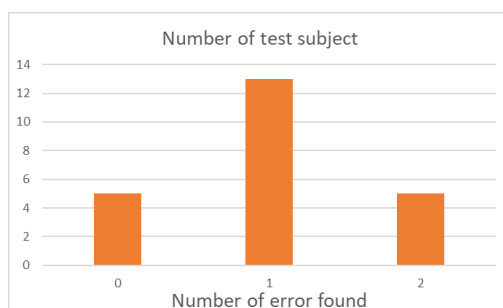


Figure 6

Results of flowchart debugging test

Based on the results, we can see that among the testing subjects, 13 persons could find only one error, 5 were able to find both of them, and the number of those who did not manage to find any was 5. Testing subjects were unable to find the index mixing-up error in most cases.

In the third phase of the investigation, testing subjects performed the Hanoi-tower test. The parameters of the Hanoi-tower specified in the PEBL test were supplemented with an additional two calculated parameters:

- 1) Compared to the lowest number of steps leading to the solution, the relative number of extra steps:

$$\text{rel_extrasteps} = (\text{steps} - \text{shortest}) / \text{shortest} = \text{extrasteps} / \text{shortest} [\%]$$

- 2) The average time between the steps during the solution:

$$\text{steps_time} = \text{time} / \text{steps} [\text{ms}]$$

The first parameter indicates that the particular testing subject attempted to solve the task in how a considered manner, with as few mistakes as possible, whilst the second indicated how quick each subject decided on the next step. The averages and standard deviations of each parameter are indicated in Table 4.

Table 4

Results of Hanoi-tower

	Shortest	steps	extrasteps	rel_extrasteps	time [ms]	steps_time [ms]
MEAN	40.3913	56.5217	16.1304	37.99%	102 041	1819.9
SD	5.72665	15.6347	11.0836	24.02%	29 335,2	270.56

The results indicate that during solving the Hanoi-tower task, the 10 tests performed could be solved in average the least 40.4 (SD=5.7) steps, whilst testing subjects managed to solve this in average 56.5 (SD=15.6) steps. The number of extra steps is thus average 16.1 (SD=11), which is relative compared to the

shortest step number 38% (SD=24%). The total time to solve the 10 tests is average 102 s (SD=29.3 s), whilst the time passed between each steps was 1.82s (SD=0.27s).

However, to perform the evaluation specified as the purpose of the test, the results of the Hanoi-Tower tests must be compared with the flowchart debugging results. The comparison may be made based on the number of errors found, rendering the Hanoi-tower test results to the number of errors found. The results are summarized in Table 5.

Table 5
Results of Flowchart error - Hanoi-tower

Flowchart	Hanoi-tower			
		extrasteps	rel_extrasteps	steps_time [ms]
0	MEAN	19.2	46.1%	1 772.0
	SD	13.6	30.9%	307.6
1	MEAN	17.4	40.3%	1 816.4
	SD	10.1	20.6%	301.9
2	MEAN	9.8	24.0%	1 877.0
	SD	11.0	24.6%	165.7

It can be seen from the results, that in case of the testing subject identifying 0 and 1 errors in the flowchart test, the extrasteps was similar, a bit higher, average 19.2 (SD=13.6) to the 0 and average 17.4 (SD=10.1) steps related to those of identifying 1 error. However, testing subjects finding both errors were able to solve the Hanoi-tower test on average less 9.8 (SD=11) extra steps. The relative number of extra steps was similar, in case of testing subjects identifying 0 error 46.1% (SD=30.9%), whilst at testing subjects identifying 1 error 40.3% (SD=20.6%). The number of extra steps was also much lower in case of those testing subjects, who managed to find both errors, 24% (SD=24.6%).

During solving the Hanoi-tower test, the mean time between the test was similar in case of testing subjects finding 0, 1 and 2 errors on the flowchart. We can learn from the results, that the mean time between the steps increased, in case of the group identifying 0 error 1.77 s (SD=0.31 s), at testing subjects finding 1 error 1.82 s (SD=0.3 s), whilst in case of testing subjects identifying both errors on the flowchart 1.88 s (SD=0.16 s).

The illustration of the results on the bar graph is indicated by Figures 7-9.

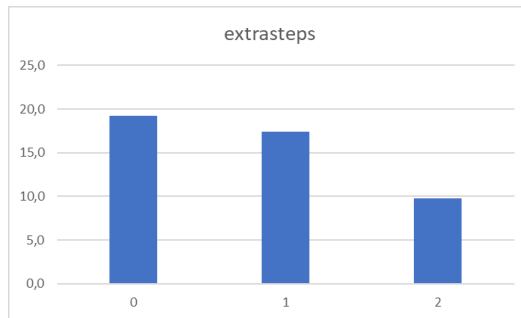


Figure 7

Identified errors in flowchart vs Hanoi-tower test extrasteps

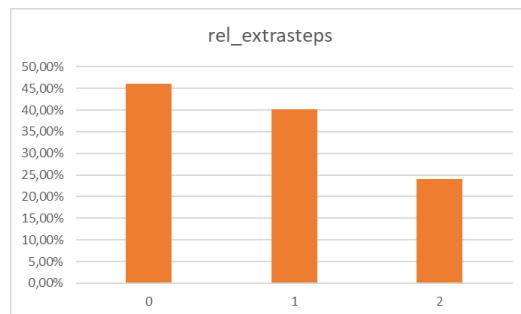


Figure 8

Identified errors in flowchart vs Hanoi-tower test relative extrasteps

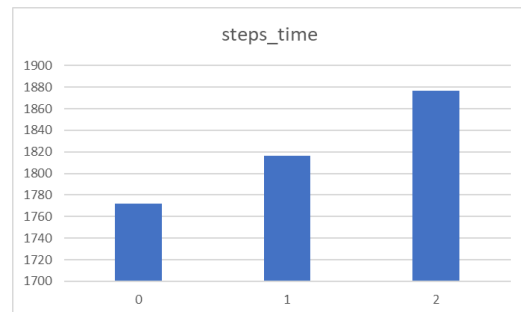


Figure 9

Identified errors in flowchart vs Hanoi-tower test step time

We can see from the results, that depending on the number of errors identified on the flowchart, if we create groups, then in case of these groups, differences can be indicated in the Hanoi-tower test results. In case of those testing subjects, who were more successful in the interpretation of the flowchart, thinking through the algorithm and as a result, identifying both errors, were able to complete the Hanoi-tower test with a better result, i.e. with less extra steps. However, it can be also

learnt, that the meant time between each step was the longest in their case. The background of the results might be that these testing subjects were thinking about the steps leading to the solution more thoroughly, thus spending more time for thinking on each step, however, they are more successful in the end, since they manage to solve the task with less extra steps. In other words, those are more successful in the end, who attempt to solve the task not by testing, rather thinking through the problem, applying a scheme on it.

Testing subjects also answered on additional questions regarding their mathematical knowledge, algorithmic abilities and language skills (1-5 Likert scale, taking the grades received in the particular subject into consideration) and regarding their previous programming knowledge, whose average results are indicated in Figures 10-12.



Figure 10

Level of math and algorithmic skills

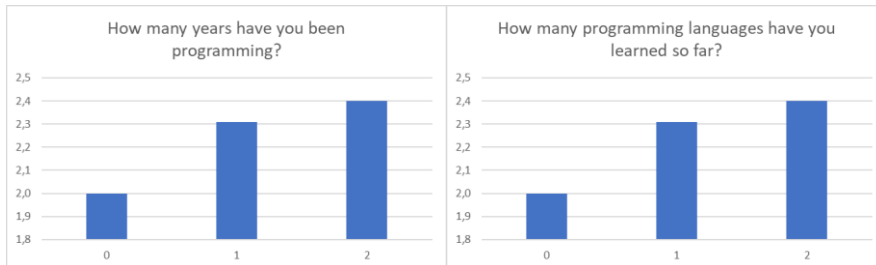


Figure 11

Prior programming experience



Figure 12

Foreign language skills

The results indicate that mathematical and algorithmic skills are in positive correlation with the success of the flowchart debug task solution. Similarly, prior programming experiences also provided a better result in solving the flowchart debug task. Interestingly, foreign language skill and programming abilities indicate a negative correlation.

Conclusions

Cognitive communication, such as, orientation, attention, memory, problem-solving and executive function, is the basis of several higher-level of cognitive skills. Executive functions control and coordinate other cognitive abilities and behavior. However, not every behavior requires control implemented by executive function, such as, reflex or other very simple, routine actions. However, if the target cannot be achieved by direct action, some kind of new behavior pattern must be established, executive functions are always necessary. The executive function, thus, plays a major role in all complex cognitive tasks, for example, problem-solving. Executive function and problem-solving driven by conscious processes are closely related to each other. Computer-based assessment of problem-solving skills and the cognitive processes it involves, is an intensively studied topic [21]-[22].

During computer software development, the determination of steps leading to the solution of the task designated as the target are developed and then coded into a series of instructions workable for the computer. In the analysis of this field, opportunities provided by cognitive info-communications [23]-[29] can be well utilized, which are applied to analyze cognitive processes in more fields [30]-[32]. The purpose of this paper was to perform tests aimed at the Executive Function and Algorithmic Thinking and to report the correlation between the two. The results indicate, that based on the tests introduced and performed in this paper, a positive correlation is indicated between the level of Executive Functions and Algorithmic Problem-solving. Better Algorithmic Problem-solving is coupled with more precise execution of Executive Functions. Results can also play an important role in increasing the efficiency of learning, through educational methods and ICT opportunities [33]-[36].

Acknowledgement

This research was supported by EFOP-3.6.1-16-2016-00003 grants, Establishment long-term R & D & I process at the University of Dunaújváros. Also thanks for the support of EFOP-3.6.2-16-2017-00018 “Produce together with the nature – agroforestry as a new out-breaking possibility” project.

References

- [1] Diamond A.: Executive functions. Annual review of psychology, 2013, Vol. 64, pp. 135-168
- [2] Zelazo P. D., Müller U.: Executive function in typical and atypical development, Blackwell handbook of childhood cognitive development, 2002, pp. 445-469
- [3] Csépe V., Györi M., Ragó A.: General Psychology 3, Osiris Kiadó, Budapest, 2008
- [4] Hughes C., Graham A.: Executive function in childhood: development and disorder, Cognitive and language development in children, 2004, pp. 205-230
- [5] Baranyi P., Csapo A., Sallai Gy.: Cognitive Infocommunications (CogInfoCom) Springer, 2015
- [6] Baranyi P., Csapó Á.: Definition and synergies of cognitive infocommunications. Acta Polytechnica Hungarica, 2012, Vol. 9, No. 1, pp. 67-83
- [7] Katona J., Ujbanyi T., Sziladi G., Kovari A.: Electroencephalogram-Based Brain-Computer Interface for Internet of Robotic Things, Cognitive Infocommunications, Theory and Applications, Springer International Publishing, 2019, pp. 253-275
- [8] Horvath I., Sudar A.: Factors contributing to the enhanced performance of the Maxwhere 3d VR platform in the distribution of digital information. Acta Polytechnica Hungarica, Vol. 15, No. 3, 2018, pp. 149-173
- [9] Guzsvinecz T., Kovacs Cs, Reich D., Szucs V., Sik-Lanyi C.: Developing a virtual reality application for the improvement of depth perception, 9th IEEE International Conference on Cognitive Infocommunications, 2018, pp. 17-22
- [10] Gilányi A., Chmielewska K.: Educational Context of Mathability. Acta Polytechnica Hungarica, Vol. 15, No. 5, 2018, pp. 223-237
- [11] Anderson S. W., et al.: Wisconsin Card Sorting Test performance as a measure of frontal lobe damage, Journal of clinical and experimental neuropsychology, 1991, Vol. 13, No. 6, pp. 909-922
- [12] Vendrell P., et al.: The role of prefrontal regions in the Stroop task. Neuropsychologia, 1995, Vol. 33, No. 3, pp. 341-352

- [13] Gomez P., Ratcliff R., Perea M.: A model of the go/no-go task. *Journal of Experimental Psychology: General*, 2007, Vol. 136, No. 3, pp. 389
- [14] Anderson J. R., Douglass S.: Tower of Hanoi: Evidence for the cost of goal retrieval. *Journal of experimental psychology: learning, memory, and cognition*, 2001, Vol. 27, No. 6, pp. 1331-1346
- [15] Merrotsy P.: The Tower of Hanoi and Inductive Logic. *Australian Senior Mathematics Journal*, 2015, Vol. 29, No. 1, pp. 16-24
- [16] Mueller S. T., Piper B. J.: The psychology experiment building language (PEBL) and PEBL test battery. *Journal of neuroscience methods*, 2014, Vol. 222, pp. 250-259
- [17] Vainio V., Sajaniemi J.: Factors in novice programmers' poor tracing skills. *ACM SIGCSE Bulletin*, 2007, 39.3, pp. 236-240
- [18] Kalat J. W.: *Biological psychology*. Nelson Education, 2015
- [19] Ericsson A.: The acquisition of expert performance as problem solving. *The psychology of problem solving*, 2003, pp. 31-83
- [20] Kovari A., Katona J. Costescu C.: Evaluation of Eye-Movement Metrics in a Software Debugging Task using GP3 Eye Tracker, *Acta Polytechnica Hungarica*, Vol. 17, No. 2, 2020, pp. 57-76
- [21] Pejić A., Stanić Molcer P.: Exploring data mining possibilities on computer based problem solving data, *Proceedings of the 14th International Symposium on Intelligent Systems and Informatics: SISY*, Subotica, Serbia, 2016, pp. 171-176
- [22] Pejić A., Stanić Molcer P.: Relationship Mining in PISA CBA 2012 Problem Solving Dataset using Association Rules, *Proceedings of the 12th International Symposium on Applied Computational Intelligence and Informatics: SACI*, Timisoara, Romania, 2018, pp. 549-553
- [23] Molnár G., Szűts, Z., Biró K.: Use of augmented reality in learning. *Acta Polytechnica Hungarica*, Vol. 15, No. 5, 2018, 209-222
- [24] Orosz B., et al.: Digital education in digital cooperative environments. *Journal of Applied Technical and Educational Sciences*, Vol. 9, No. 4, 2019, pp. 55-69
- [25] Horvath I.: MaxWhere 3D Capabilities Contributing to the Enhanced Efficiency of the Trello 2D Management Software. *Acta Polytechnica Hungarica*, Vol. 16, No. 6, 2019, pp. 55-71
- [26] Sziládi G.: Applications of human-computer interfaces and related psychological-educational aspects. *Computers & Learning*, Vol. 2, No. 1, 2019, pp. 1-14
- [27] Nagy K., et al.: Gamification and microcontent orientated methodological solutions based on bring-your-own device logic in higher education, *IEEE*,

- Proceedings of the 9th IEEE International Conference on Cognitive Infocommunications: CogInfoCom, Budapest, 2018, pp. 385-388
- [28] Námesztovszki Zs, et al.: Research of the usage of health-behavior improving mobile applications, Proceedings of the 10th IEEE International Conference on Cognitive Infocommunications: CogInfoCom, Naples, 2019, pp. 619-624
- [29] Ósz R., András I., Rajcsányi-Molnár M.: Educational management questions of new generation smart phones: mobile generation in a changing learning environment – pedagogic challenges and paradigms (Az újgenerációs mobil oktatásszervezés kérdései: A mobil generáció a változó tanulási környezetben – pedagógiai kihívások és paradigmák). In: András I., Rajcsányi-Molnár M. (eds.): Metamorphosis: Glocal Dilemmas in Three Acts (Metamorfózis: Globális dilemmák három tételben). Új Mandátum Kiadó, Budapest, 2013, pp. 196-215
- [30] Jambor S.: Educational methods based on student activity in vocational education. Transactions on IT and Engineering Education, Vol. 2, No. 1, 2019, pp. 17-29
- [31] Rajcsányi-Molnár M. MaTech: Creative math competition for secondary school students assessing digital device usage (MaTech: Digitális eszközhasználaton alapuló kreatív matematika verseny szervezése középiskolás tanulóknak). In: Fodorné T. K. (ed.): Higher education innovations in the era of learning: challenges of digitalization, skill development and networking (Felsőoktatási innovációk a tanulás korában: a digitalizáció, képességfejlesztés és a hálózatosodás kihívásai), MELLearn Egyesület, Pécs, 2019, pp. 19-32
- [32] Gogh E., Kovari, A.: Experiences of Self-regulated Learning in a Vocational Secondary School. Journal of Applied Technical and Educational Sciences, 9(2), 2019, 72-86
- [33] Benedek A. et al.: Practices of Crowdsourcing in relation to Big Data Analysis and Education Methods, IEEE 13th International Symposium on Intelligent Systems and Informatics, Subotica, Serbia, 2015, 167-172
- [34] Biró K., et al. The effects of virtual and augmented learning environments on the learning process in secondary school. 8th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Debrecen, Hungary, 2017, 371-376
- [35] Kovari A., Katona J., Costescu C.: Quantitative Analysis of Relationship Between Visual Attention and Eye-Hand Coordination, Acta Polytechnica Hungarica, Vol. 17, No. 2, 2020, pp. 77-95
- [36] Pinter R. et al.: Massive Open Courses (MOOCs) held in Serbian. Journal for Information Technology Education Development and Teaching Methods of Technical and Natural Sciences, Vol. 6, No. 1, 2016, 12-16