EJSBS
The European Journal of Social & Behavioural Science
ISSN: 2301-2218 (ONLINE)
OPEN ACCESS

The European Journal of Social and Behavioural Sciences
EJSBS Volume XXVII, Issue I (eISSN: 2301-2218)

COMPARATIVE ANALYSIS OF SURVEYS FOCUSED ON PUPILS' ATTITUDES TOWARD THE TEACHING OF PROGRAMMING IN SCHOOLS

Tomáš Hornik[a]

[a]*University of Hradec Králové, Rokitanského 62, Hradec Králové 500 03, Czech Republic*

Abstract

The teaching of programming has been implemented at elementary schools worldwide as early as at the age of six. Even though the subject matter in the teaching of programming is mostly given, the specific form and content of lessons can significantly influence the attitudes of pupils, yet there is no standardized tool for evaluation of pupils' attitude toward the course. This study aims to compare and analyze existing attitudes surveys specifically focused on programming while considering limitations of secondary school pupils (ISCED-2). The article is based on theoretical and methodological literature review and a comparative analysis of existing surveys. Current surveys are mostly focused on university students and the questions are far too complex to be understood by pupils at elementary schools, making it necessary to create and validate new attitudes survey purely for the purpose of the topic of programming, that would reflect secondary school pupils limitations. Such a survey could be further used to compare different programming courses and their impact on pupils' attitudes. The research serves as a foundation for the development and testing of a new attitudes survey derived from existing ones and specifically modified for secondary school pupils.

Keywords: Teaching programming, pupils' attitudes, survey research

*Corresponding author.
E-mail address: tomas.hornik@uhk.cz

doi: 10.15405/ejsbs.270

Received: 17 September, 2018; **Revised:** 19 November, 2020; **Accepted:** 13 January, 2020

https://doi.org/10.15405/ejsbs.270
eISSN: 2301-2218 / Corresponding Author: Tomáš Hornik
Selection & Peer-review under responsibility of the Editors

1. Introduction

The ever-rising level of technology present in everyday life has led to a greater demand in specific areas of education, most obviously, in the subject of information technology and computer science. Both pupils at elementary schools and students at high schools should be studying technology more deeply and even though, they may not all become professional programmers, they should have at least the most elemental grasp of basic concepts that anchor the technological world surrounding us. The most visible change in modern curricula connected with this fact is the shift from IT to Computer Science and introduction of the most basic programming concepts starting as early as on ISCED level 1 (International Standard Classification of Education) with a strong emphasis on the topic especially at ISCED levels 2 and 3, which can be seen for example in Britain (Kemp, 2014, p.6), the USA (The White House, 2016), the Slovak Republic (Štátny pedagogický ústav, 2014) and also in the Czech Republic (Národní ústav pro vzdělávání, 2018).

The terminology used in the paper is based on the Czech school system, where elementary schools consist of primary (ISCED-1, pupils aged 6 to 10) and secondary schools (ISCED-2, from age 11 to 14) and as such when discussing teaching of programming in general, the term elementary school is used. However, the research in this paper is focused on evaluating attitudes surveys for use specifically on secondary schools.

Despite the fact, that the subject of programming has been taught at universities for decades, there is still a severe lack of IT experts (Poskočilová, 2018) and the subject's complexity has led to high drop-out rate, averaging at 67% pass rate worldwide (Watson & Li, 2014). These issues were targeted by multiple studies (Ford & Venema, 2010; Pejcinovic Holtzman, Wong, & Recketenwald, 2017) which conclude that there were various different approaches to resolve the problem. So far, there is no definitive ideal method of teaching programming that would mitigate said problems, in fact, it might not be necessary at all. Aforementioned changes in curricula have also caused a significant shift in the target audience age group. The introduction to the topic has moved from universities to high schools and in recent years to elementary schools, thus preparing children for complex IT and computer science topics from early age. This will in the long-term naturally solve problems with the high drop-out rate, yet, at the same time such a radical lowering of the target age group will necessarily create problems of a different kind.

2. Problem Statement

Major changes connected with said shift in the age group is related to the fact that while most high schools and all universities are selective in student population intake, elementary

https://doi.org/10.15405/ejsbs.270
eISSN: 2301-2218 / Corresponding Author: Tomáš Hornik
Selection & Peer-review under responsibility of the Editors

schools are open to the general population. This means that every child is going to be exposed to the topic to a certain degree since the topic is no longer voluntary, but mandatory. It can be stated with certainty based on long-term teaching practice, that in such a case, there will always be a group of children with attitudes ranging from indifference to even loathing for the topic of programming, which can cause resistance or even defiance, thus effectively disrupting the education process itself. Disruptive behavior resulting from these attitudes of even a single pupil has a negative impact on the rest of the class and hinders the progress and depth to which the topic can be presented in the course of a lesson. As such, it is imperative to focus not only on academic performance, but also on pupils' attitudes.

Attitudes are defined by Průcha, Mareš, and Walterová (2003) as "evaluative relations taken by an individual on the outside world, other subjects and themselves. It includes the disposition to behave or react in a specific relatively stable way" (p.317). Albarracin, Sunderrajan, Lohmann, Chan, and Jiang (2018) state in *The Psychology of Attitudes, Motivation and Persuasion*, that the definition must be sufficiently comprehensive and generalizable while still taking into consideration the modern trends. In their research, they concluded that "what has been consistent in the multiple conceptualizations of the attitude construct is that evaluation is the key component" (p. 4) and based on that simplified the overall definition to a statement that attitudes are evaluation.

3. Research Questions

Even though attitudes are only one of the variables affecting the pupils' study achievements, their positive impact has been well documented (Shabbir, Asif, & Saeed, 2015; Narmadha & Chamundeswari, 2013) and cannot be overlooked. The importance of attitudes rises as age of pupils becomes lower. The combination of a complex topic, such as programming, and large classes of young non-selective pupils necessitates adjustments in the style of teaching. However, in order to make these adjustments, the teacher needs a tool that would make attitudes assessment possible.

Programming has been taught for decades at the universities and because of the problems stated in the introduction of this paper, there are already validated attitude surveys tested on large samples. This article attempts to answer following questions:

3.1 Are current validated attitude surveys focused on attitudes toward programming suitable for secondary school pupils (aged 11 to 14)?

3.2 What are the problems and limitations related to the use of already existing attitude surveys in regular teaching practice when teaching programming at secondary schools?

4. Purpose of the Study

Currently, there is a large number of different teaching tools, languages and environments specifically focused on teaching basic programming concepts to complete beginners (e.g. Hour of Code, Scratch, Logo, Turtle Academy, Udacity, CodeCombat, etc.) and some of these projects can be used by pupils themselves without a teacher because of their tutorial nature. Most of the curricula (see introduction) dictate only a general guideline which leaves the teachers to choose the particular tool and approach on their own. These approaches may vary widely, from using a textbook or computer-guided tutorials with the teacher working only as a supporting element to the teacher-centered instruction in the frontal lecture style and everything in between. No matter what approach the teacher chooses, the outcome must be in compliance with the curriculum thus ensuring that all the specific topics are being taught. Nonetheless, ensuring only the content is transmitted might not be enough and the teacher may choose a wrong tool for the wrong class, leading to boredom and disengagement of the pupils.

The ideal attitude survey for secondary school pupils should be robust enough for any and all programming courses. It should give the teachers an option to evaluate how their lessons are perceived by pupils and act accordingly. Such a survey should allow for a comparison of different courses and, with a sample large enough, also objectively state which courses are perceived better by the pupils. The purpose of this study is to determine whether there already is an attitude survey suitable for teaching programming at the secondary school and if not, serve as an entry point for a new one.

5. Research Methods

This article is the first part of the larger research focused on establishing and testing of a new tool for studying pupils' attitudes towards the topic of programming in general as well as towards the specific programming course used in their lessons. As such, the basic research method is the literature review followed by the comparative analysis of selected surveys. This provides ground work for future survey development and testing with regard to the attitudes towards programming, including specific methodology and tools used for already established and validated surveys.

Sources for the literature review were limited by certain criteria, whereas the main limiting factors were source language and accessibility. Source data type were not considered as a limiting factor and the literature review includes research papers, articles from both journals and proceedings, surveys, master's and doctoral dissertations, university textbooks, scientific monographs, books and websites. However, regarding the source accessibility, only the databases that are either free and publically available or accessible under the university

https://doi.org/10.15405/ejsbs.270
eISSN: 2301-2218 / Corresponding Author: Tomáš Hornik
Selection & Peer-review under responsibility of the Editors

credentials were used, such as Google Scholar, Web of Science, SCOPUS, ScienceDirect, ResearchGate, etc. As for the source language, the limitation was put on the use of only Czech, Slovak and predominantly English language. Time frame was not set due to the fact that a large number of important and high impact articles on the topic of attitudes dates back to 1980s, but there was a clear effort to use articles about teaching the programming dating no more than 10 years back and preferably post-dating 2014 (a sort of a breaking point with the introduction of the Hour of Code).

For the search in the databases, the following key words were used: teaching programming, pupils' attitudes, attitudes and achievements, attitudes survey, survey research, educational programming languages. After the first preliminary selection, which eliminated irrelevant articles, 176 closely relevant sources were found. Attitudes surveys focused on sciences and math were still regarded as relevant because of their close relationship in the type of mental faculties similar to the ones necessary for the understanding of programming.

Based on the literature research, eight surveys (presented in section 5.2 below) were selected for the focused overview in the comparative analysis part. Wider literature research included also the topic of attitudes, their influence and changes in general as well as cognitive development of children and adolescents in order to set specific parameters that must be taken into consideration while working with an age specific group like secondary school pupils.

5.1. Establishing essential parameters

The difference between university freshman students and the secondary school pupil is massive even merely from the biological point of view, where the prefrontal cortex (which, among other things, affects the ability to focus attention, considering multiple information in complex situations, organizing thoughts and solving problems, considering and planning future, etc.) matures throughout the whole adolescence which spans ages 10 to 24 (Arain et al., 2013).

Dumontheil (2014) claims, that the period of adolescence, defined in his study as the ages between 10 and 19, is "an important period of development in terms of the acquisition of higher cognitive skills" (p. 58). With regard to the development of rostral prefrontal cortex (RPFC) Dumontheil (2014) states, that "the lateral parts of RPFC (RLPFC) appear to support the ability to detach oneself from the environment and to elaborate, evaluate and maintain abstract rules and information, as it is involved in reasoning, problem solving, and more generally abstract thinking" (p. 59). Since he also proved with magnetic resonance imaging that this is almost the last part that matures during the development in adolescent period (and in general), the cognitive abilities of children in early and late adolescence are unquestionably

different. All attitude surveys require the respondent to think about their opinions on something, to evaluate some phenomena, and since "RLPFC is thought to be specifically involved in the elaboration, evaluation and maintenance of abstract rules" (Dumontheil, 2014, p. 59), it cannot be reasonably expected of children to answer the same questions with the same quality as adults even if it is only about their own opinions.

Based on the aforementioned cognitive development, changes in adolescent brain and experience from teaching practice, the following criteria were established for considering an attitude survey as suitable for the secondary school pupils:

- The questions must *be simple and short enough to be fully understood by children as young as 11 years because of their limited cognitive abilities.*
- *The number of questions should be reasonable (max. 25) due to the limited attention span.*
- *The overall time necessary for finishing the whole survey should be around 15 to 20 minutes for it to be usable in standard teaching practice.*
- *The survey must not be focused on a specific programming language or environment.*

5.2. Selected attitudes about programming surveys

The following eight surveys were found based on the literature research that focused on pupils' attitudes towards programming. As a lot of research has been done on attitudes towards mathematics, some of the following programming surveys are based on attitudes to maths. Although they were not excluded from the literature review as such, they are not part of the following summary of programming surveys. Further information concerning the surveys studying the pupils' attitudes towards programming are presented in section 6 below.

https://doi.org/10.15405/ejsbs.270
eISSN: 2301-2218 / Corresponding Author: Tomáš Hornik
Selection & Peer-review under responsibility of the Editors

Table 1. Summary of surveys on attitudes towards programming

Survey and its author(s)	Number of items	Item types	Content structure	Survey testing	Sample size	Commentary
Impact on Attitudes and Self-Efficacy with CS by Phillips & Brooks (2017)	**4**	four-point Likert scale	pre-test and post-test with one HoC lesson; items divided into two factors – "attitude" and "self-efficacy"	paired *t*-test	**8040** elementary and middle school students	focused on Hour of Code, conducted in Dec. 2016
Attitudes toward Learning Programming through Visual Interactive Environments by Asad, Tibi, & Raiyn (2016)	**29**	five-point Likert scale	items in three categories – "motivation," "competition" and "challenge"; survey accompanied by a written exam, assignments, and final projects	descriptive statistics, Cronbach's Alpha from 0.69 to 0.99	**24** elementary school pupils	based on Baser (2013); used three different learning environments
Attitudes towards computer programming and knowledge of programming by Du, Wimmer & Rada (2016)	**4** (on attitudes)	four and five-point Likert scale	participant information followed by only four attitude items; also tested programming comprehension; used pre-test and post-test	descriptive statistics with two-tailed paired *t*-test	**116** university students	focused on Hour of Code; similar to this paper author's master's thesis
Attitudes Evaluation Tools by Klement Klement & Lavrinčík (2012)	**12** (students) and **16** (teachers)	dichotomic questions (yes-no)	different survey for students and teachers; items divided into 7 factor areas	descriptive statistics only	**321** grammar school students and **12** teachers	for testing course in Visual Basic 2012; only in Czech lang.
Attitude Scale of Computer Programming Learning (ASCOPL) by Korkmaz & Altun (2014)	**20**	five-point Likert scale	two-pool indication between positive-ness and negative-ness; factor "Willingness" with 9 positive items, factor "Negativity" with 6 items and remaining items are "Necessity"	exploratory and conf. factor analyses, item factor correlations, corrected correlations, etc.	**496** university students in the first phase and **262** in the second	fully validity and reliability tested survey
Computing Attitudes Survey by Tew, Dorn & Schneider (2012)	**10** (on attitudes, overall length 53)	five-point Likert scale	measures novice-to-expert attitude shifts with items divided into eight factors (only the last factor "Personal Interest & Enjoyment" is attitude oriented)	exploratory and confirmatory factor analyses, KMO = 0.866 and Bartlett's test $\chi^2(595) = 335$ 9	**447** university students	fully validity and reliability tested; based on Colorado Learning Attitudes about Science Survey (CLASS)
Programming Attitude Scale by Baser (2013)	**35**	five-point Likert scale	items divided into four factors (see chap. 6)	Exploratory factor analysis, reliability from 0.87 to 0.93	**179** university students	based on Wiebe (2003)
Computer Science Attitude Survey by Wiebe, Williams, Yang & Miller (2003)	**57**	five-point Likert scale	questions divided into five subscales	Cronbach's alpha from 0.83 to 0.91	**162** university students	based on Fennema-Sherman mathematics attitude scale

https://doi.org/10.15405/ejsbs.270
eISSN: 2301-2218 / Corresponding Author: Tomáš Hornik
Selection & Peer-review under responsibility of the Editors

6. Findings

Of all the eight selected surveys, none was ascertained as completely ideal for large scale testing on secondary schools. Reasons for this decision are given in this section along with the promising ideas and findings that could influence results even on secondary schools. A survey developed by Peluso and Sprechini (2012) regarding the attitudes towards programming (tested as a part of two-week course on high-schools in North central Pennsylvania with total sample size of 70 students) was omitted even from the table itself since the questionnaire is not available to the public, number of attitudes items was stated as "several questions" (page 7) and only five of them can be inferred from presented charts. Another problem with this survey for the purposes of this paper is its focus on programming language Alice, which is specifically named in several of these questions. A large number of attitude surveys focused on the topic of computer science in general, were all omitted except Wiebe et al. (2003) since this survey was used as a direct cornerstone by Baser (2013) and is described alongside it.

An in depth analysis and comparison of all the selected surveys are presented in the following two sub-sections using a point-by-point scheme in the first sub-section, and text-by-text organization in the second sub-section because of the dissimilarity of surveys. Each survey is compared with the set of established parameters from section 5.1 and further advantages and disadvantages of the given survey are stated. Sub-section 6.1 compares two surveys (accompanied by the author's Master's thesis), that were considered as unsuitable for the use at an elementary (secondary) school but share the same structure and methods. Sub-section 6.2 describes the remaining surveys also unsuitable for elementary schools which greatly vary in research methods.

6.1. Surveys with similar research methods and organizational structures

Most similarities are shared by surveys from Du et al. (2016) and Phillips and Brooks (2017). Both contain only four attitude questions and both researches were based on a pre-test, sample Hour of Code online lesson followed by a post-test. The same organizational scheme and methodical approach was also used by the author of this paper in his Master's thesis (Hornik, 2016). All three researches unfortunately provide a very limited, shallow and incomplete picture of pupils' attitudes and are considered as overly simplified. The questions are short enough, however in case of Phillips and Brooks (2017), the use of the term "computer science" may be confusing for some pupils. Otherwise, all three surveys comply with the set of parameters from sub-section 5.1.

https://doi.org/10.15405/ejsbs.270
eISSN: 2301-2218 / Corresponding Author: Tomáš Hornik
Selection & Peer-review under responsibility of the Editors

Despite the difference in respondent groups, Du et al.'s (2016) was focused on university students, Hornik's (2016) on elementary school pupils and Phillips & Brooks's (2017) on elementary and middle schools students), Du et al.'s (2016) and Hornik's (2016) came to the same observation. Hour of Code definitely has a positive impact on attitude but there was no significant change in programming skills. Possible reasons for such phenomena are discussed in the said researches. Phillips and Brooks's (2017) survey did not consider the factor of improving students' programming skill at all.

The research data from all three surveys were analyzed mostly only with descriptive statistics methods, which were complemented with two-tailed paired-sample *t*-test in case of Du et al.'s (2016) and Phillips and Brooks's (2017), whereas Kendall rank correlation coefficient and qualitative interviews were used in case of Hornik's (2016) Master's thesis. Du et al. (2016) encountered problems connected with their attempt to combine four and five-point Likert Scales; Hornik (2016) had to convert different types of variables to ordinal scales and Phillips & Brooks's (2017) use the term "Computer Science" in all their questions instead of "programming" despite the fact that the core of the lesson is about programming and algorithmization. Since children associate new information only in connection with presented material, there is a high likelihood that they understood all these terms as synonyms even though they most certainly are not.

Phillips and Brooks's (2017) research was not published in a peer-reviewed journal but only as a supportive study directly from the project Hour of Code. Even though they provided complete raw data publically available for further studies (a direct link is included in the paper), the research summary is somehow lacking by using incomplete data *"8,040 students completed at least one question on both the pre-and post-Hour of Code surveys, and 48% (3,891) reported gender"* (page 1). Nevertheless, this study has by far the largest data sample of all the selected surveys.

Creators of the Hour of Code project, *Code.org*, created not only a single-lesson tutorial, but several complete curricula. As a part of their Computer Science Principles curriculum, they created a much larger Pre-Survey containing 31 questions of dichotomic type, six-point Liker Scale type ordinal variables, nominal variables and even open-ended questions. Unfortunately, I was not able to find any comprehensive article published by anyone from Code.org that summarized collected data and presented any overall results. Nonetheless, Code.org has a large number of ongoing projects and such a study might be made publically available in the future.

Very similar to aforementioned surveys is the research of Asad, Tibi, and Raiyn (2016), which is also the only survey focused purely on elementary school pupils. The authors claim

their goal was to create an exploratory model of pupils' attitudes towards learning computer programming that would support the choice of visual learning environments. The number and complexity of questions is appropriate so the questions are easy to understand. The survey was presented to the pupils after three weeks of ICT lessons during which three different visual programming environments were presented. The researchers found out that the motivation and competition levels increased while the level of challenge remained same. The biggest drawback of this study is its very small scale (24 respondents) and a limited data analysis which cannot statistically substantiate any of the claims, thus the research should be considered in terms of a qualitative case study.

6.2. Surveys considered unsuitable for use on ISCED level 2 schools

Klement, Klement and Lavrinčík (2012) created two evaluation tools (one survey for pupils and one for teachers) intended for testing a new electronic textbook of programming in a grammar school. The textbook, along with all the supplementary materials, is publically available in the Czech language at http://www.pros.upol.cz/vyuka-programovani. Questions are short and simple and the number of items is appropriate. Unfortunately, it is the only questionnaire that does not use Likert scale items, but only dichotomic yes-no questions. This choice presents no problems for simple course evaluation tool, but dichotomic questions severely limit precision of results as well as use of more complex statistical methods.

Klement et al. (2012) found that the topic of programming was significantly more interesting for boys (page 54) and that 71.16 % of students consider the textbook suitable, but only 17.55 % would like to continue with programming lessons.

Tew, Dorn and Schneider (2012) discovered that there is no validated and reliable survey for testing attitudes towards computer science and for this purpose they modified the Colorado Learning Attitudes about Science Survey (CLASS), which has already been successfully modified for biology and chemistry. The original CLASS questions were at first revised and adjusted for the topic of computer science and both faculty experts and students were interviewed in order to confirm consistent interpretation of questions. Questions were afterwards assessed by 37 experts and based on their consensus, fourteen questions were excluded. Further specific steps for the survey modification are described in detail in their research paper. In the 2012 version, the questionnaire had 53 questions with 10 of them focused on attitudes. The latest version (Dorn & Tew, 2015) has been further reduced to 26 questions of which only four (Q4, Q11, Q13 and Q26). are about personal attitudes. The survey is fully validated and tested and the main approach is to measure novice-to-expert attitude shifts, i.e. the depth and complexity of respondents' understanding of the topic. Both

this approach and the questions are completely unsuitable for elementary schools. Very important for research on elementary school pupils would be Q19 *"We use this statement to discard the surveys of people who are not reading the questions. Please select "Agree" for this question to preserve your answers."* since in previous research some pupils were not reading the questions at all, choosing answers randomly, which they also admitted in the qualitative part of the study.

Even though the full version of the survey from Korkmaz and Altun (2014) is not publically available, it is the first fully validated and reliability tested survey focusing strictly on the topic of programming. The paper does not contain data evaluation and describes solely the validation and reliability testing conducted on the survey itself. The paper provides the exact wording of the survey items albeit with crude English translation containing a large number of mistakes. The questions are meant strictly for the university students and cannot be used on an elementary school.

Baser (2013) studied the attitudes of future Information and Computer Technology teachers towards the topic of programming and for this purpose, he developed a new questionnaire based on Wiebe (2003). Wiebe's (2003) original 57 questions were modified and reduced by Baser (2013) to at first 47 and in the final phase into 35 questions. At the end of the process, Baser (2013) states that his questions have minimal similarity with Wiebe's (2003). The whole process is described in detail in Baser's (2013) paper. The final 35 questions were divided by Scree Plot into four factors called *"Confidence in learning computer programming; Usefulness of computer programming; Attitudes toward success in computer programming;* and *Effective motivation in computer programming."* Unfortunately, the full version of the survey is not publically available and as such suitability for elementary school pupils could not have been effectively concluded. Interesting findings were that the future ICT teacher attitudes towards programming were not negative, but also not very positive with a mean score of attitudes at only 3.59. As in the other studies, female attitude scores were significantly lower than male.

Wiebe et al. (2003) based his survey on the fully validated Fennema-Sherman Mathematics Attitudes Scale and even though it is labeled as Computer Science Attitude Survey, it was used in a study focused on pair programming (Williams, Wiebe, Yang, Ferzli, & Miller, 2002). Surprisingly, no difference was detected in attitudes between solo and pair programming. Even though Wiebe et al.'s (2003) survey is not validated and the focus group was again university students, the questions are mostly simple and could be in theory adapted for secondary school pupils.

7. Conclusion

Despite the shift to teaching introductory programming courses to elementary and secondary schools, there is still no reliability and validity tested survey for measuring attitudes towards the learning of programming among this particular target age group and towards the given course that would be usable at the secondary school. The only questionnaires with items that could be used as a ground for creation of a new survey for secondary school pupils are from Asad et al. (2016), Klement et al. (2012) and Wiebe et al. (2003). Most of the surveys are focused on university students (Korkmaz & Altun, 2014; Baser, 2013; Tew et al., 2012, Wiebe et al., 2003) and the questions cannot be applied on elementary school pupils.

The development of a new survey will necessarily reflect findings from the whole literature review and certain ideas will need to be implemented in the survey creation. In order to evaluate the impact of a given course on pupils, there need to be a pre-test accompanied by a post-test (as used by Du et al. (2016), Phillips and Brooks (2017) and Asad, Tibi and Raiyn (2016). An option of a survey for teachers, similar to Klement et al.'s (2012) also needs to be considered. It can be further extended to allow comparison of teachers' opinions on pupils' attitudes and the real findings based on pupils' responses. Limiting factors are based on cognitive development of pupils and were established according to biological brain development (Arain, 2013; Dumontheil, 2014), teaching practice and curricular needs.

Even though the survey is not meant to be a gender study, findings from all the surveys point out significant difference in attitude towards programming between male and female respondents. Further literature research should be, at least partially, oriented in this way in order to reflect these findings. This concludes the first phase of larger research and will be followed by development and testing of a new survey that would reflect all the findings presented in this paper.

Acknowledgements

The paper has been supported by the Specific Research Project of the Faculty of Education, University of Hradec Králové, 2019. The author(s) declare that there is no conflict of interest.

References

Albarracin, D., Sunderrajan, A., Lohmann, S., Chan, S., & Jiang, D. (2018). The Psychology of Attitudes, Motivation, and Persuasion. In D. Albarracin & B. T. Johnson (Eds.), *Handbook of Attitudes, Volume 1: Basic Principles* (2nd ed., pp. 1-105). New York: Psychology Press. https://doi.org/10.4324/9781315178103

Arain, M., Haque, M., Johal, L., Mathur, P., Nel, W., … & Sharma, S. (2013). Maturation of the adolescent brain. *Neuropsychiatr Dis Treat., 9,* 449-61. https://doi.org/10.2147/NDT.S39776

Asad, K., Tibi, M., & Raiyn, J. (2016) Primary School Pupils' Attitudes toward Learning Programming through Visual Interactive Environments. *World Journal of Education*, *6*(5), 20-26. https://doi.org/10.5430/wje.v6n5p20

Baser, M. (2013). Attitude, Gender and Achievement in Computer Programming. *Middle-East Journal of Scientific Research, 14*, 248-255. Retrieved from https://files.eric.ed.gov/fulltext/ED542330.pdf

Dorn, B., & Tew, A. E. (2015). Empirical validation and application of the computing attitudes survey. *Computer Science Education*, *25*, 1-36. https://doi.org/10.1080/08993408.2015.1014142

Du, J., Wimmer, H., & Rada, R. (2016). "Hour of Code": Can it Change Students' Attitudes toward Programming? *Journal of Information Technology Education: Innovations in Practice, 15,* 52-73. https://doi.org/10.28945/3421

Dumontheil, I. (2014). Development of abstract thinking during childhood and adolescence: The role of rostrolateral prefrontal cortex. *Developmental Cognitive Neuroscience*, *10*, 57-76. https://doi.org/10.1016/j.dcn.2014.07.009

Ford, M., & Venema, S. (2010). Assessing the success of an introductory programming course. *Journal of Information Technology Education: Research*, *9*(1), 133-145.

Hornik, T. (2016). *Možnosti rozvíjení algoritmického myšlení s využitím projektů Hour of Code a Scratch* [The Possibilities of Development of Algorithmic Thinking with Projects Hour of Code and Scratch], Hradec Králové: Pedagogical Faculty University of Hradec Králové. 135 s. Unpublished Master's Thesis.

Kemp, P. (2014). *Computing in the National Curriculum: A Guide for Secondary Teachers*. Bedford: Newnorth Print, Ltd. Retrieved from https://www.computingatschool.org.uk/data/uploads/ cas_secondary.pdf

Klement, M., Klement, J., & Lavrinčík, J. (2012). *Metody realizace a hodnocení výuky základů programování*. [Methods of Realization and Evaluation of Teaching of Programming Foundations] Olomouc: 2012.

Korkmaz, Ö., & Altun, H. (2014). A validity and reliability study of the Attitude Scale of Computer Programming Learning (ASCOPL). *Mevlana International Journal of Education*, *4*(1), 30-43. https://doi.org/10.13054/mije.13.73.4.1

Narmadha, U., & Chamundeswari, S. (2013). Attitude towards Learning of Science and Academic Achievement in Science among Students at the Secondary Level. *Journal of Sociological Research, 4*(2), 114-124. https://doi.org/10.5296/jsr.v4i2.3910

Národní ústav pro vzdělávání: Metodický portál RVP. (2018). *RVP v oblasti informatiky a ICT* [Framework Education Programme in the Field of Computer Science and ICT] [online]. Praha: Výzkumný ústav pedagogický v Praze. Retrieved from http://www.nuv.cz/t/revize-rvp-ict

Pejcinovic, B., Holtzman, M., Wong, P. K., & Recketenwald, G. (2017). Assessing student preparedness for introductory engineering and programming courses. https://doi.org/10.1109/FIE.2017.8190539

Peluso, E. M., & Sprechini, G. (2012). The impact of Alice on the Attitudes of High School students Toward Computing. *Journal for Computing Teachers, 7*, 2-11.

Phillips, R. S., & Brooks, B. (2017). The Hour of Code: Impact on Attitudes Towards and Self-Efficacy with Computer Science. Retrieved from https://code.org/files/HourOfCodeImpactStudy_Jan2017.pdf

Poskočilová, M. (2018). "Ajťáci" chybějí dvěma třetinám firem ["IT Guys" Are Missing in Two Thirds of the Companies] *Statistika & My: Měsíčník Českého statistického úřadu* [online]. Praha: Český statistický úřad. Retrieved from http://www.statistikaamy.cz/2018/10/ajtaci-chybeji-dvema-tretinam-firem/

Průcha, J., Mareš, J., & Walterová, E. (2003). *Pedagogický slovník*. 4. aktualizované vydání. [Pedagogical Dictionary] Praha: Portál.

Shabbir, M. A., Asif, I., & Saeed, M. A. (2015). Students' Attitude towards Science and its Relationship with Achievement Score at Intermediate Level. *Journal of Elementary Education, 25*(2), 61-72.

Štátny pedagogický ústav (2014). *Inovovaný ŠVP pre 2. stupeň ZŠ – Informatika* [Innovated School Educational Programme for Secondary Schools – Computer Science] [online]. Bratislava: ŠPÚ. Retrieved from http://www.statpedu.sk/files/articles/dokumenty/inovovany-statny-vzdelavaci-program/informatika_nsv_2014.pdf

Tew, A. E., Dorn, B., & Schneider, O. (2012). Toward a validated computing attitudes survey. In: Proceedings of the Ninth Annual International Conference on International Computing Education Research (p. 135-142). Available at https://dl.acm.org/citation.cfm?id=2361303

The White House, Office of the Press Secretary (2016). *FACT SHEET: President Obama Announces Computer Science For All Initiative* [online]. Washington, DC: January 30, 2016. Retrieved from https://www.whitehouse.gov/sites/whitehouse.gov/files/images/FACT%20SHEET%2BPresident%20Obama%20Announces%20Computer%20Science%20For%20All%20Initiative_0.pdf

Watson, C., & Li, F. W. B. (2014). Failure rates in introductory programming revisited. In: *Proceedings of the 2014 conference on Innovation & technology in computer science education (ITiCSE '14)* (pp. 39-44). New York: Association for Computing Machinery. https://doi.org/10.1145/2591708.2591749

Wiebe, E., Williams, A. L., Yang, K., & Miller, C. (2003). Computer Science Attitude Survey. Retrieved from https://www.researchgate.net/publication/248503161_Computer_Science_Attitude_Survey

Williams, L., Wiebe, E., Yang, K., Ferzli, M., & Miller, C. (2002). In Support of Pair Programming in the Introductory Computer Science Course. *Computer Science Education*, *12*(3), 197-212. https://doi.org/10.1076/csed.12.3.197.8618