THE TEACHING OF MATHEMATICS

2009, Vol. XII, 1, pp. 25–31

PREDICTORS OF SUCCESS IN SOLVING PROGRAMMING TASKS

Jasmina Milić

Abstract. Based on a sample of 79 third-grade high school students, this study found that success in solving programming tasks at higher level of knowledge depends on success in solving programming tasks at lower level of knowledge, intellectual abilities of students and their computer attitudes. Success in solving programming tasks at lower knowledge level did not depend on their intellectual abilities, computer attitudes and gender.

ZDM Subject Classification: Q24, Q74; *AMS Subject Classification*: 00A35.

Key words and phrases: Knowledge level; intellectual ability; gender; computer attitude; programming.

Introduction

So far research related to defining the level of knowledge has made use of various classifications. Levels of knowledge have been defined by some models of modified Bloom taxonomy of knowledge [1] or SOLO taxonomy [3, 12].

This study used a didactical model derived from Bloom taxonomy [7], which singles out four levels of knowledge:

1. Run—performance of completed program,
2. Read—reading of program code,
3. Change—change of program code, and
4. Create—creating of program.

In our research we studied two levels of knowledge, which are, according to our pedagogical experience, critical for success in solving programming tasks. These levels were:

1. Read—lower level of knowledge, and
2. Change—higher level of knowledge.

This study not only studied success in solving programming tasks at both above levels, but it also examined the relation between these two successes.

Success in learning of programming should be examined in relation to cognitive abilities and mode of thinking (as in studies [15, 16]. According to our reading, studies to date have not examined this relation for various levels of knowledge, which was thus done in this research.

Which other variables might be the factors of success in solving programming tasks?

One of these variables is computer attitude. This is because this attitude influences not only acceptance of computers, but also their utilization as professional tools or assistants in learning and teaching [17]. As boys usually have more positive computer attitude than girls [4, 9], it is possible that gender is also one of variables influencing success in solving programming tasks, especially in view of research [13] finding that abstract of "hard" programming style is reserved for males, whereas actual or "soft" programming style is reserved for females.

Taking into consideration the research context mentioned above, it is important to study whether cognitive abilities, computer attitude and gender are the factors of success in solving programming tasks at the two knowledge levels selected above. Thus, in this research we studied the following two research questions:

1. Is success in solving programming tasks at the lower knowledge level dependent on intellectual abilities, computer attitude and gender?
2. Is success in solving programming tasks at the upper knowledge level dependent on success in solving lower level programming tasks, intellectual abilities, computer attitude and gender?

Method

Sample and procedure

The sample included three 11th grade classes oriented toward studying mathematics and science in XIV Belgrade Gymnasium (high school). There were 79 students: 38 boys and 41 girls. The actual number of students of in these three classes was higher (91), but only students with complete data on all examined variables were included in this study. As all tested students were taught by the same teacher, the teacher's influence on motivation and success was equal for all of them.

The knowledge test was developed according to the requirements from the *Curriculum of Computer Science and Information Technology for 11th (3rd) Grade of High Schools* (for the total number of 30 lessons). The test included all the units related to Pascal-based programming covering the subject matter for the first semester of 3rd grade.

Empirical research was performed in the second semester of the 2007/2008 school year. The knowledge test was compulsory for all students. The intelligence test and computer attitude survey were voluntary, and they were carried out by the school psychologist. The overall research was performed under the conditions free from copying and consulting among students.

Instruments and variables

The developed knowledge test included two levels of knowledge. At lower level, by the means of five tasks, students were required to read the program code and answer the question as to what was gained by its execution. At higher level of knowledge, by the means of another five tasks (with the content corresponding to that of the lower level ones), students were required to change or improve the given program code to attain the expected execution. The test was evaluated by two independent evaluators in the way that the first one evaluated each test, whereas the second one did so for every third test. The correlation between the two evaluators' scores was acceptable. The lowest correlation for individual task scores was 0.82; the highest was 0.96. As regards the two evaluators' overall scores, the correlations were 0.94 for the lower and 0.95 for the higher level of knowledge. The contents of the test and the evaluation key are given in the Appendix.

General intellectual abilities were tested by S-1 specialization test, from KOG3 cybernetic battery of intelligence tests [14], whereas computer attitude was assessed by Selwyn's computer attitude scale [8].

The variables used in this research were:

Knowledge 1—the student's success in solving lower level programming tasks. Minimum and maximum values of this variable were 0 and 5.

Knowledge 2—the student's success in solving higher level programming tasks. Minimum and maximum values of this variable were also 0 and 5.

IQ—the student's intellectual abilities. Minimum and maximum values of this variable were 0 and 30.

Attitude—the student's computer attitude. Minimum and maximum values of this variable were 21 and 105.

Gender—the student's gender (1 for male and 2 for female).

Results

Table 1 shows the mean values (M), standard deviations (SD), reliabilities (α) and correlation matrix for the examined variables.

Variable	M	SD	α	2	3	4	5
1. **Knowledge 1**	3.54	1.46	0.77	**0.567****	0.049	0.161	0.112
2. **Knowledge 2**	1.54	1.36	0.80		**0.227***	**0.284***	−0.069
3. **IQ**	22.13	6.15	0.90			0.041	**−0.241***
4. **Attitude**	84.75	7.70	0.78				−0.072
5. **Gender**†	1.52						

* $p < 0.05$, ** $p < 0.01$, † 1–male, 2–female

Table 1. Descriptive statistics and correlation analysis

Due to statistically significant correlations between variable **Knowledge 2** and variables **Knowledge 1**, **IQ** and **Attitude**, a stepwise linear regression (pin=0.05) with **Knowledge 2** as the dependent variable and **Knowledge 1**, **IQ** and **Attitude** as the three predictors was performed. All these predictors remained in the regression equation respectively explaining 32.1%, 4% and 3.6% (39.7%) variance of the dependent variable.

Discussion

Taking into consideration the research questions stated at the end of Introduction, the following findings emerged:

1. Success in solving programming tasks at lower level of knowledge (Read) could not be related to students' intellectual abilities, computer attitudes or gender.
2. Success in solving programming tasks at higher level of knowledge (Change) could be explained via success in solving programming tasks at lower level of knowledge (Read), students' intellectual abilities and computer attitudes.

Considering the complexity of programming tasks at lower level of knowledge (Read), the first finding was more or less expected. It complies with the findings of [11], which underlines that students' intellectual abilities are brought to prominence in tasks requiring superior forms of learning. This reference [11] also emphasizes that the results achieved at lower level of knowledge reveal more about student's readiness to accept the obligation to study certain subject, than his/her talent or interest in the subject. This should apply for computer attitude, which was not found in this research. However, when the relevant correlation was corrected for measurement error [10], it may be taken that success in solving programming tasks at lower level of knowledge correlates with students' computer attitudes ($r = 0.208$, $p = 0.066$; $p < 0.05$ for one-tailed test).

The second finding confirms that success in programming at higher level of knowledge depends not only on success in programming at lower level, but also on students' intellectual abilities and their computer attitudes. This finding is in accord with previous studies that found that higher levels of knowledge require lower levels of knowledge [5], and that higher level programming requires higher cognitive abilities [16]. The finding also points out the importance of positive computer attitude in learning programming [8, 17]. Contrary to our expectation, gender was not related to success in solving programming tasks at higher level of knowledge. This might be because there was correlation between variables **Gender** and **IQ** ($r = -0.241$, $p < 0.05$), which probably contributed to the lack of correlation between variables **Knowledge 2** and **Gender**. However, this explanation is not supported by the relevant data, because partial correlation between variables **Knowledge 2** and **Gender**, controlling for variable **IQ**, is marginal ($r = -0.015$, $p = 0.896$).

Since predictors **Knowledge 1**, **IQ** and **Attitude** jointly explained only 39.7% of the variance of dependent variable **Knowledge 2**, there are other predictors of programming success at **Change** level. Psychologists often refer to internal and external motivation as factors of (un)success [11]. According to [2, 3, 12], the mode of learning may be an important predictor of the success in question. In addition to students' general intellectual abilities and computer attitudes, students' creativity, success in solving mathematical tasks and ability of solving problems may be predictors of programming success [6]. Our experience suggests that computer experience in programming should also be considered as relevant factor. Further research may thus be aimed at studying success in programming at higher knowledge level by using these and other relevant predictors.

REFERENCES

1. Anderson, L. W., Krathwohl, D. R., Airasian, R., Chruikshank, P.W., Mayer, K. A., Pintrich, R. E., et al., *Taxonomy for Learning, Teaching and Assessing, A Revision of Blooms Taxonomy of Educational Objectives*, New York, Addison Weslay Longman, 2001.
2. Biggs, J., *Teaching for Quality Learning at University*, Buckingham, UK, SRHE and Open University Press. 1999.
3. Biggs, J., Collis, K., *Evaluating the Quality of Learning: the SOLO Taxonomy*, New York, Academic Press, 1982.
4. Brosnan M., *The role of psychological gender in the computer-related attitudes and attainments of primary school children (aged 6–11)*, Computers & Education, **30** (1998), 3–4, 203–208.
5. Cox, K., Clark, D., *The use of formative quizzes for deep learning*, Computers & Education, **30**, 3/4 (1997), 157–167.
6. Erdogan, Y., Aydin, E., Kabaca, T., *Exploring the psychological predictors of programming achievement*, J. Instructional Psychology, **35**, 3 (2004), 264–270.
7. Kaasboll, J. J., *Exploring didactic models for programming*, University of Oslo, Department of Informatics, 1998.
8. Kadijevich, Dj., *Representativity, reliability, homogeneity and validity of Selwyn's computer attitude scale for 16–19 education*, Psihologija, **33**, 3/4 (2000), 491–498.
9. Kirkpatrick, H., Cuban, L., *Should we be worried? What the research says about gender differences in access, use, attitudes and achievement with computers*, Educational Technology, **38**, 4 (1998), 56–61.
10. Osborne, J. W., *Effect sizes and the disattenuation of correlation and regresion coefficients: Lessons from educational psychology*, Practical Assessment, Research & Evaluation, 8, 11 (2003). Internet: `http://pareonline.net/getvn.asp?v=8&n=11`
11. Smiljanić, M., *Certain factors of (un)success in college studies of students with exceptional high school success* (in Serbian). University of Belgrade, Faculty of Philosophy, 1976.
12. Thompson, E., *Holistic assessment criteria—Applying SOLO to programming projects*, Conferences in Research in Practice in Information Technology (Vol. 66), 2007. Internet: `http://crpit.com/`
13. Turkle, S., *Women and computer programming: A different approach*, Technology Review, **87**, 8 (1984), 49–50.
14. Volf, B., Momirović, K., Džamonja, Z., *KOG3 – Intelligence Test Battery* (in Serbian), Društvo psihologa Srbije – Centar za primenjenu psihologiju, Beograd, 1992.
15. White, G., *Visual Basic programming impact on cognitive development of college students*, J. Information Systems Education, **17**, 4 (2006), 421–427.
16. White, G., Ploeger, F., *Cognitive characteristics for learning Visual Basic*, J. Computer Information Systems, **44**, 3 (2004), 58–66.
17. Woodrow, J., *A comparison of four computer attitude scales*, J. Educational Computing Research, **7**, 2 (1991), 165–187.

XIV Belgrade Gymnasium

E-mail: `jagomil@nadlanu.com`

APPENDIX—Knowledge test and evaluation key

Question	Answer	POINTS

1. What results from the execution of the following program for input value of **25?**
`program p1;`
`var i, j, k : integer;`
`begin`
`read(i);`
`j := i div 7;`
`k := i mod 7;`
`writeln(j, k);`
`end.`
	Answer (in view of program execution):
	`i = 25`
	`j = 25 div 7 = 3`
	`k = 25 mod 7 = 4`
	This program outputs the following values:
	`j = 3`
	`k = 4`
	(*Correct answer, but without such a description of the program execution: ½ point.*)
		
		1/2
		1/2

2. What is the output for the execution of the following commands?
`a := 9; b := 5;`
`if (a < b) then`
`begin`
`d := sqr(a - b);`
`end`
`else`
`begin`
`d := sqrt(a - b);`
`end;`
`writeln('d =', d);`
	Answer (in view of program execution):
	`a = 9, b = 5`
	`? 9 < 5 - no`
	`d = sqrt(9 - 5) = 2`
	`d = 2`
	(*Correct answer, but without such a description of the program execution: ½ point.*)
		1

3. Which value of variable **m** is printed by the following program for input values **i = 3** and **j = 4**?
`program p3;`
`var i, j, n, m : integer;`
`begin`
`read(i,j);`
`m := 0;`
`for n := i to j do`
`m := m + n;`
`writeln('m =',m);`
`end.`
	Answer (in view of program execution):
	`i = 3, j = 4, m = 0`
	`n = 3, m = 0 + 3 = 3`
	`n = 4, m = 3 + 4 = 7`
	`m = 7`
	(*This answer is only accepted with such a description of the program execution.*)
		1

4. What results from the execution of the following program?
`program p4;`
`var m, n, i, b : integer;`
`begin`
`m := 2; n := 7; b := 0;`
`for i := m to n do`
`if (i mod 2 = 0) then b := b + 1;`
`writeln('b =', b);`
`end.`
	Answer (in view of program execution):
	`m = 2, n = 7, b = 0`
	`i = 2 ? 2 mod 2 = 0, yes, b = 1`
	`i = 3 ? 3 mod 2 = 0, no`
	`i = 4 ? 4 mod 2 = 0, yes, b = 2`
	`i = 5 ? 5 mod 2 = 0, no`
	`i = 6 ? 6 mod 2 = 0, yes, b = 3`
	`i = 7 ? 7 mod 2 = 0, no`
	`b = 3`
	(*This answer is only accepted with such a description of the program execution.*)
		1

5. What results from the execution of the following program for input sequence **10, 17, 8, 21** and **-99** entered via the keyboard?
`program numbers;`
`var x, a : integer;`
`begin`
`a := 0;`
`readln(x);`
`while x <> -99 do`
`begin`
`if x > a then a := x;`
`readln(x);`
`end;`
`writeln('a =', a);`
`end.`
	Answer (in view of program execution):
	`a = 0`
	`x = 10`
	`? 10 <> -99, yes ? 10 > 0, yes, a = 10`
	`x = 17`
	`? 17 <> -99, yes ? 17 > 10, yes, a = 17`
	`x = 8`
	`? 8 <> -99, yes ? 8 > 17, no`
	`x = 21`
	`? 21 <> -99, yes ? 21 > 17, yes, a = 21`
	`x = -99`
	`? -99 <> -99, no`
	`a = 21`
	(*This answer is only accepted with such a description of the program execution.*)
		1

6. Change and improve the given program code to have time interval given in minutes shown in hours and minutes.
 	
`program p6;`
`var Min, H, M : integer;`
`begin`
`write('Time in minutes: ');`
`readln(Min);`
`H := Min mod 60;`
`writeln(Min, ' min. =', H, ' h. and', M, ' min.');`
`end.`
	Answer:
	
	`program p6;`
	`var Min, H, M : integer;`
	`begin`
	`write('Time in minutes: ');`
	`readln(Min);`
	`H := Min div 60;`
	`M := Min mod 60;`
	`writeln(Min, ' min. =', H, ' h. and', M, ' min.');`
	`end.`
		1/2
		1/2

7. The following program calculates height **ha** of isosceles triangle with base **a** and side **b**. Improve this program to calculate and print this height when **a** and **b** are positive integers satisfying relation **b > a/2**. Otherwise, print a comment that the input is not correct.
 	
`program p7;`
`var a,b,ha: real;`
`begin`
`read(a, b);`
`ha:= sqrt(sqr(b)-sqr(a/2));`
`writeln('ha =', ha);`
`end.`
	Answer:
	
	`program p7;`
	`var a, b, ha : real;`
	`begin`
	`read(a,b);`
	`if b > a/2 then`
	`begin`
	`ha := sqrt(sqr(b)-sqr(a/2));`
	`writeln('ha =', ha);`
	`end`
	`else`
	`writeln('Wrong input valus!');`
	`end.`
		1/4
		1/4
		1/4
		1/4

8. Below is given a part of program that calculates the sum of the first **n** members of sequence **1 + 1/2 + 1/3 + 1/4 + ...** Change and upgrade the program code to calculate the sum of the first **n** numbers.
 	
`program p8;`
`var n, i : integer; s : real;`
`begin`
`...`
`s := s + 1/i ;`
`...`
`end.`
	Answer:
	
	`program p8;`
	`var n, i : integer; s : real;`
	`begin`
	`read(n);`
	`for i := 1 to n do`
	`s := s+i;`
	`writeln('s =', s);`
	`end.`
		1/4
		1/4
		1/4
		1/4

9. Modify the program below to have it print the number of two-digit numbers divisible by the sum of their own digits (e.g. **42** is divisible by **4+2**).
 	
`program p9;`
`var a, b, c, i, num : integer;`
`begin`
`num := 0;`
`a := i div 10;`
`num := num + 1;`
`writeln('num =', num);`
`end.`
	Answer:
	
	`program p9;`
	`var a, b, c, i, num : integer;`
	`begin`
	`num := 0;`
	`for i := 10 to 99 do`
	`begin`
	`a := i div 10;`
	`b := i mod 10;`
	`c := a + b;`
	`if (i mod c = 0) then num := num + 1;`
	`end;`
	`writeln('num =', num);`
	`end.`
		1/4
		1/4
		1/4
		1/4

10. Modify the code below to have it determine the lowest summer temperature. This temperature is determined when 0 is entered for daily temperature.
 	
`program p10;`
`var min, t : integer;`
`begin`
`min := t;`
`writeln('Enter daily temperature: ');`
`read(t);`
`while`
`...`
`writeln('Lowest temperature is:', min);`
`end.`
	Answer:
	
	`program p10;`
	`var min, t : integer;`
	`begin`
	`writeln('Enter daily temperature');`
	`read(t);`
	`min := t;`
	`while t <> 0 do`
	`begin`
	`if t < min then min := t;`
	`writeln('Enter daily temperature');`
	`read(t);`
	`end;`
	`writeln('Lowest temp. is:', min);`
	`end.`
		1/4
		1/4
		1/4
		1/4