Progress Log for Mentoring Programming Education

Zsuzsanna Szalayné Tahy[1] and Zoltán Czirkos[2]

[1] Eötvös Loránd University, Faculty of Informatics
Budapest, Hungary
[2] Budapest University of Technology and Economics, Department of Electron Devices
Budapest, Hungary
sztzs@caesar.elte.hu, czirkos@eet.bme.hu

Abstract. Programming knowledge has become necessary for people. There are several methods describing how to teach programming but the success of the usage of these methods depends on the students' preliminary knowledge and the teachers' attitude. Moreover, the groups are usually not separated by knowledge level or skills and are therefore very heterogeneous. Every student is different and has their own way to learn programming but the key stages and exams are fixed. Our research focuses on monitoring the personal development of programming knowledge in a big group. The main questions are: how can students prepare themselves in an effective way; how can teachers notice problems before the exams; how can the curriculum developer utilize the feedback. We propose a tool named the Progress Log to solve these problems and present the experience we gained by introducing it in a course in which six hundred students take part.

Keywords: Teaching-learning process, Programming, Computational thinking, Creativity

1 Introduction

There are many studies monitoring groups of students being taught programming using new methods. Several studies try to find the best programming language or the best topics, tasks, concepts to base the curriculum and other knowledge on them [2, 7]. These studies conclude that the 'new method is more effective than the old one' for the analysed group.

The BME IT Engineering education[8] combines more methods from the ones presented in [6] as it develops practical programming knowledge. The main preliminary skills are the practice in programming, Mathematics and Physics. The professional expectations have high priority, but the personal skills of students are the key to success. The preliminary knowledge and attitude of the students are not negligible and teaching the usability of concepts is more important than the contents of the course or the amount of concepts [5].

In order to measure the learning process, we developed a novel tool to monitor students' progress during the semester. We call it the Progress Log. It is an expanding questionnaire that is the first task in the class each week. In this log, the students provide self-assessments about their knowledge of various topics in the course material. By constantly monitoring their answers, we can analyse 1) how students use log, 2) whether it helps them learn and 3) whether the choices in the log – or seeing that students update their previous entries in the log – help teachers mentor the learning process of the students.

2 Pedagogical Background

One of the main goals of pedagogical and psychological studies is to understand the learning process. Models based on that research help recognize the typical situations and problems in everyday teaching practice.

2.1 The Learning Activity Unit

In order to explore the learning process, we have defined a learning model in our previous article [4]. The unit of the model is the Learning Activity Unit (LAU) which describes the learning of a concept or a skill as well as a course unit.

The Learning Activity Unit encompasses the complete learning cycle, from the starting point, when the students first meet a new concept, to the point when they are able to use it on their own. The phases defined by the model are as follows:
1. Initial learning ⇒ **2. Trying** ⇒ **3. Experimenting** ⇒
4. Pause ⇒
5. Using {**a) Repeating**; **b) Modifying**; **c) Creating** ⇒ Back to phase 4.

The LAU-Model describes the system of the LAUs. The model – with the relationships between LAUs – describes a complex learning process. The quality of a course depends on the model: the LAUs should follow the aim of the course, should consistently build upon each other and should be detailed enough.

A teaching-learning process can be read in [3], which is based on the Use–Modify–Create model of Lee [1]. The process is denoted by the letters PRIMM (Predict, Run, Investigate/Explain, Modify, Make/Create/Design). PRIMM is very similar to the LAU, the phases of the models can be paired: Predict–**Initial learning**, Run–**Trying**, Investigate–**Experimenting**, Modify–**Modifying**, Make–**Creating**. The more relevant difference is the role of **Pause** which is very important in learning effectiveness. On the one hand, it implies the **Repeating**, but on the other it is required for a proper **Creating** phase.

The **Pause** seems to be harmful but it is necessary. The learning process seems to be finished when the student can use the knowledge in a given situation. While passing the tests the student archives all the things she learned to passive knowledge. Some pause and a new situation is needed to reach the **Creating** phase. The creative use of knowledge means that the new task is not aimed specifically at that knowledge but using it is a part of the solution. LAU-5c, **Creating** means that the knowledge or skill is recallable and usable in other contexts.

3 The Progress Log as a Curriculum Development Tool

In order to solve the learning problems caused by long pauses and memorizing tasks in self-education, we created the Progress Log to be used in our own course.

3.1 Organizational Background

Basics of Programming 1 is one of the main courses for first year students of software engineering education at BME. The number of students is 550–600 but there are plans to expand it to 800. The course design is centralized, there is a course coordinator who organizes the training, compiles the series of tasks and tests. The employment of teachers is based on professional knowledge. Lecturers have doctorate degree, the trainers on lab have at least BSc diploma, and the lab classes are helped by demonstrators too. Educators' pedagogical knowledge is based on their own experience, formal training of pedagogy or teaching methodology is not expected.

The programming course went through a redesign in 2015, as we presented it on ISSEP'2016 [4, 8]. The most important change was the reorganizing of the order of lesson types (lecture, laboratory work and practice) to fit the LAU-Model much more closely than the original: **Initial learning** in the lecture, **Trying** during lab classes and **Experimenting** in practice lessons. According to the expectations of the course, the **Pause** and **Using** phases are part of self-education. Although students have some homework, they are expected to do self-organized practice.

3.2 The Progress Log

The Progress Log is our tool for students to improve their programming skills. It is an expanding questionnaire that is the suggested first task in lab classes. Each lab practice of the students started with the task of refreshing the Progress Log. Each item of the log is named after a major concept of the curriculum and there are six possible statements (answers) about student's knowledge. Every item offers a choice no. "0", that expresses "I don't know anything about it". Choice no. "1" corresponds to LAU **Initial learning** and **Trying**; no. "2" describes practice in the LAU **Experimenting**. Choice no "3" shows that LAU **Repeating** has been done. Choice no. "4" describes a modified usage of the concept while no. "5" gives a new approach where the concept at hand is a tool for **Creating** a new one. Usually the higher level of usage is associated with a concept that will be presented later, and this drives the student to do some creative thinking instead of simply archiving the new knowledge. When students first meet the item in the laboratory classes, our expected answer is at least no. "3". Reaching no. "4" is recommended on the first evaluation of knowing the concept.

The Progress Log aims to be a guide for the students to find out the required levels of knowledge for each topic of the course. We have to be aware that motivating the student without telling the solutions is much more difficult than re-teaching. The Progress Log does not contain any memorisable templates, it is not a list of problems already solved, it is rather a dictionary of terms. This way Progress Log does not re-teach but rather motivates to recall and improve knowledge.

Teachers were informed that they can follow the answers. It was declared that neither answers are used in any evaluations, nor is answering mandatory. Teachers are allowed to suggest the use of Progress Log for the student and they can suggest tasks when seeing low values. There was no training about pedagogical background nor about LAU and LAU-Model.

4 The Impact of Using Progress Log

4.1 Global Aspects of Using Progress Log

The analysed group size was 617 but 56 students did not answer at all (9%). All the 18 topic were answered by 377 students (61%). About 20% of the students stopped using the progress log shortly after the first Midterm Test. Only 7% of the students rated themselves much worse than the results showed and 0.5% rated themselves well but the evaluations were very poor.

The most remarkable result is that 51% of the students (317 out of 617) updated their answers in the log during the second half of the semester for some of the first 8 topics. This unexpectedly high activity means that students roll back to older topics, and this gives them a chance to recall the concepts learned earlier. Moreover, this activity shows a gamification effect. The increasing level of the answers at each topic serve as level of a game, and students are motivated to get to a higher rated choice – and realize what they have achieved.

Progress Log clarifies aims and helps to get a better result. Having compared the results of the courses of the last two years we saw that enrolled students' knowledge level is lower in the analysed year (2016) than a year before but the average of final marks is higher (see Table 1).

Table 1. Characteristics of courses started in 2015 and 2016

	2015	**2016**	**trend**
Number of students	565	617	↗
The average of points in university admission	424.65	423.87	↘
The minimum of points in university admission	375	370	↘
The average of answers in preliminary knowledge survey	4.02	3.44	↘
The average of results (final marks in 0–5 scale)	2.88	2.94	↗

The proportion of students' results shows the details of the higher average of the marks. 70% of the students completed the course from low (2 – satisfactory) to high (5 – excellent) level in both years. We can see that the number (and the proportion) of satisfactory result decreased while excellent results increased in 2016. In the meanwhile, the proportion of students rated is higher among the failed students. Both changes in proportions show that the Progress Log helps students to learn more purposefully and, therefore, effectively.

Analysing the results of each evaluation, the success rate raises significantly in some Small Tests. The first Small Test resulted more flawless paper in 2015 than in 2016. The second and fifth Small Tests resulted in similar scores in 2015 and 2016, but the third and fourth tests were definitely more successful in 2016 than in 2015. The results of the sixth test show that students try to optimize their score. This test is considered in the final result only if it is better than the fifth one: the final mark could depend on a few points. For those students who had to get those points, the Progress Log was a help to work more effectively.

4.2 The Break of Progress

Mark 0 has a special meaning, because it is not a real value, it indicates those students who dropped out before the end of the semester. The number of these students were 76 in 2015 and 77 in 2016. Analysing their very poor achievement we have to see that there is a group of students who cannot finish the course. The proportion of these students is smaller in 2016 than in 2015 but they realize earlier that the expectations exceed their capabilities. These students selected low level choices in the Log and most of them skipped the log weeks before the final result. We can say that 2/3 of those students who abandon to log their progress have problems with requirements.

The missing choices are not the only signals of the students having problems with the progress. A student updating the choices in a negative direction gives an alarm signal to the teacher. It means that the student thinks that he has the expected knowledge, but he is disappointed after failing a test: he cannot recall the knowledge at a critical moment. The reason we have seen is that the student copies the code instead of memorizing, or memorizes a concept instead of applying it in a different context.

4.3 Improvement of the Curriculum

Progress Log follows the concept of LAU and the curriculum is designed according to the LAU-Model. The evaluation of the curriculum consists of two main parts: the students' results and the usability of taught knowledge. Correlating every (18) topic of the Progress Log to the (8) evaluations gives a useful tool to analyse the inside structure of the curriculum and the evaluation system. The relative high or low correlations show the global and abnormal features of the explored course which is useful for curriculum developer:

1. The last topics have a very small correlation to evaluations. This is normal for any course since there is no time left to practise the last topics. There are only low correlations in the rows of topics taught earlier (eg. debugging technique). That raises a question how it could be evaluated.
2. The Small Tests focus on one or two topics. It is expected that the correlations of these topics to the STs are the maximum among the correlation of the Small Test.

3. Some Small Tests show high correlations with topics not being assessed. These denote the dependencies between LAU-s. The topics regarding the abstraction and the indirection require a higher level of cognitive skills.
4. A Test with low correlations to every topic indicates the need of further research.

5 Future Development

The Progress Log has a potential for students to understand the quality of the knowledge, the expectations of using the concept before the exams. On the other side, the Progress Log helps educators to detect problems in knowledge and in learning methods as well. Mentoring could become more effective while students improve their self-study skills. By analysing the Progress Log, target-oriented tutoring could be organised including the learning methods and the development of skills.

As students seem to be willing to update their previous choices, it is worth monitoring the trends and variations as well. Attention should be paid to negative modification, permanently low levels and missing choices, so that the teacher can help the student to catch up before it is too late.

References

1. Lee, I., Computational Thinking for Youth in Practice. In: ACM Inroads vol: 2 no: 1, pp: 32–37, ACM. DOI: 10.1145/1929887.1929902 https://users.soe.ucsc.edu/~linda/pubs/ACMInroads.pdf (2011)
2. Malan, D. J., Leitner, H. H.: Scratch for Budding Computer Scientists. In: Proceedings of the 38th SIGCSE Technical Symposium on Computer Science Education, ACM Covington Kentucky USA, pp: 223–227. DOI: 10.1145/1227310.1227388 http://doi.acm.org/10.1145/1227310.1227388 (2007)
3. Sentance, S.: Exploring pedagogies for teaching programming in school, http://blogs.kcl.ac.uk/cser/2017/02/20/exploring-pedagogies-for-teaching-programming-in-school (2017)
4. Szalayné Tahy, Zs., Czirkos, Z.: "Why Can't I Learn Programming?" The Learning and Teaching Environment of Programming. In: Brodnik A., Tort F. (eds) ISSEP 2016. LNCS vol 9973. Springer, Cham DOI: 10.1007/978-3-319-46747-4_17 (2016)
5. Szalayné Tahy, Zs., Czirkos, Z.: The Two Worlds of Programming In: Stoffová V., Horváth R. (eds.) XXXth DidMatTech 2017. pp:59–67 ISBN: 978-80-568-0029-4 (2017)
6. Szlávi, P. and Zsakó, L.: Methods of teaching programming 1(2). In Teaching Mathematics and Computer Science, 1.02, pp. 247-258. Univ. of Debrecen, Hungary (2003)
7. Uolevi, N., Orleana, G., Jussi, K.: A Motivation Guided Holistic Rehabilitation of the First Programming Course. CM Transactions on Computing Education, vol. 11, no. 4 (2011)
8. Basics of Programming 1, Software Engineering Course Datasheet, BME Faculty of Electrical Engineering and Informatics https://portal.vik.bme.hu/kepzes/targyak/VIEEAA00/en/; live course site: https://infoc.eet.bme.hu/