Xia, Belle Selene

An In-Depth Analysis of Learning Goals in Higher Education: Evidence from the Programming Education

Published in:
JOURNAL OF LEARNING DESIGN

DOI:
10.5204/jld.v10i2.287

Published: 01/01/2017

Document Version
Publisher's PDF, also known as Version of record

Please cite the original version:
Xia, B. S. (2017). An In-Depth Analysis of Learning Goals in Higher Education: Evidence from the Programming Education. JOURNAL OF LEARNING DESIGN, 10(2), 25-34. DOI: 10.5204/jld.v10i2.287

An In-depth Analysis of Learning Goals in Higher Education: Evidence from the Programming Education

Belle Selene Xia
Department of Information and Computer Science
Aalto University, Finland
selene.xia@aalto.fi

Abstract

Previous research has shown that, despite the importance of programming education, there is limited research done on programming education experiences from the students' point of view and the need to do so is strong. By understanding the student behaviour, their learning styles, their expectation and motivation to learn, the quality of teaching can be improved. The goal of this paper is to examine the connection between educational theories and student-centred pedagogy via an empirical study. While research results have confirmed student difficulties in learning programming in terms of the retention and completion rates of the programming courses, we will propose some of the solutions to overcome these challenges. We will also classify the various definitions of learning goals both theoretically and empirically in order to further our understanding in the subject field. New research opportunities are opened in the applied work of a personalised learning environment.

Keywords

Educational approaches, educational theory, quality of teaching, learning goals, programming education

Introduction

Lister and Leaney (2003) suggested that, from the teacher's point of view, one of the primary goals of programming courses is to equip students with the skill to develop programs to solve computing problems expressed both in programming and non-programming terms. From the student's point of view, learning goals reflect individual motivation (Pintrich, 2003; Xia & Liitiäinen, 2014). Goals are set in order to reach a specific performance outcome. In programming education, having clearly defined goals motivates learning and thus enables successful learning outcomes. (Locke & Latham, 2006) As a matter of fact, learning outcomes can be defined in general as acting as a benchmark for ensuring teaching quality (Maher, 2004; Xia, 2013).

Bruse et al. (2004) defined learning as deepening one's personal experience of a given phenomenon and teaching as enhancing the students' experience of the given phenomenon through the alignment of critical dimensions in these experiences. Specifically, the primary goal of teaching in introductory programming education is to support learners to learn programming concepts and obtain useful programming skills. However, previous research has demonstrated a high student failure rate in terms of the course completion and retention rates as a major concern of these programming courses (Bruse et al., 2004). Some of the challenges associated with the programming courses are:

- *Difficulties in reading and writing program codes:* In order to be able to write correct program codes, students need to be able to first read program codes, which is one of the main challenges of programming education especially for those with no prior computer science background (Buck & Stucki, 2000).

- *Difficulties in designing program statements:* Programming students fail to learn to write and design programs as programming skills are cognitively complex skills that require a deep understanding of structurally complex content (Dehnadi & Bornat, 2006).

- *Difficulties in tracing the program codes:* Research results have shown that a large group of students could not trace through the programs in a systematic manner upon request. This phenomenon is especially evident among the novice programmers (Lister & Leaney, 2003).

- *Difficulties with grasping programming concepts:* Misconceptions exist related to the architecture of computers and semantics students found difficult to grasp, such as what the assignment statement in Java does, understanding of recursion as a program construct, of what an object is and of how a given C program executes (Booth, 1992).

Robins (2010) emphasised that the quality of learning can be improved by setting specific and measurable learning goals. These goals are especially important in programming education and there is a pressing need for further research (Krathwohl, 2002). The goal of this paper is to evaluate how students' background influences their attitude towards programming and their learning preferences via an empirical study. The results of the present study will add to instructors' and researchers' knowledge in establishing a more personalised learning environment for the students in higher education. This paper is organised as follows. We will introduce our conceptual framework for this study based on the literature review on learning goals in programming education after defining the objective of this study. Then, we will elaborate on the data and method used in this study. Thereafter, we will discuss our research results and link their impact with the previous literature on the subject matter.

The research question in this study is formulated as follows: *What are the different factors that affect the kind of learning goals students set for themselves in programming education?* The aim of this research is to deepen our knowledge in the field of programming pedagogy through an in-depth analysis of students' experience in programming education.

Background

In 1956, Benjamin Bloom established the following six levels of taxonomy from lowest to the highest that is still widely used in understanding student learning of programming in terms of learning goals of the cognitive branch of the taxonomy. In programming education, Bloom's Taxonomy is used to motivate improvements to the quality of the programming courses and understand the learning of programming along with the appropriateness of different assessment forms (Bloom, Engelhart, Furst, Hill, & Krathwohl, 1956). In particular, Bloom's Taxonomy was created to emphasise that learning goals should be set at all level of the taxonomy and at different stages of learning. The Bloom original taxonomy included:

- **Knowledge**: the student can remember, name and define specific facts or methods.

- **Comprehension**: the student can interpret, explain and discuss the meaning of certain concepts or facts.

- **Application**: the student can solve problems and produce solutions by applying knowledge to new situations.

- **Analysis**: the student can break down, analyse and infer information into meaningful pieces in order to determine the motives or causes of their relationships.

- **Synthesis**: the student can create, compose and invent new information from the existing knowledge.

- **Evaluation**: the student can evaluate, appraise and compare different arguments in light of a set of criteria in order to judgment of the given information.

Code-tracing skills, for instance, are represented in Bloom's Taxonomy through *comprehension* (being able to understand) and *analysis* (being able to analyse program-writing assignments of different kinds). As another example, the ability to write program codes in problem solving belongs to the *synthesis* level of the Bloom's Taxonomy. The revised Bloom's Taxonomy (composed of a revised hierarchy of create, evaluate, analyse, apply, understand and recall) (Anderson et al., 2001) is used as a guideline by computer science studies, such as ACM (Association for Computing Machinery) and IEEE (IEEE Computer Society), for developing their computer science curriculum and programming education in general. Use of Bloom's Taxonomy ensures that learning goals set in the introductory programming courses are cognitively challenging. Given that developing the skill of being able to read and design programs is a key focus of computer science education, it is likely that some proportion of the high failure rate in introductory programming courses may be attributed to the difficulty in setting optimal learning goals and appropriate assessment instruments which outlines the importance of this study.

Conceptual Framework

Figure 1 summarises learning goals from both the student's and the teacher's point of view based on the previous. Defining learning goals is used to enhance learning outcomes and motivate learning. For them to be effective, the implementation of these goals should be monitored and controlled according to the course demand and learning contexts. In some cases, learning goals are used interchangeably with performance goals and learning outcomes. Research results show that the student's age, prior experiences and attitude towards learning, preference in the course delivery format and the average study period are the main explanatory factors behind different learning outcomes (Lim & Morris, 2009; Xia, 2013; Xia & Liitiäinen, 2016).

Figure 1. Factors that affect learning goals

The different aspects of learning goals in programming education are:

- *Specific, task-involved:* Specific, proximal and challenging goals enhance self-efficacy and improved performance (Eccles & Wigfield, 2002).

- *General, broadly defined:* While learning goals are found to have an effect on performance, depending on the task complexity, one could emphasize the broader context of programming languages and the programming community. In this case, learning goals can be general and broadly defined (Nahrgang et al., 2013).

- *Academic:* Learning goals play a decisive role in determining the quality of learning. In order to affect the learning performance of the students, different levels of academic goals can be set depending on the task in question, the learning environment and the learning beliefs (Valle et al., 2009).

- *Context-dependent:* Factors that contribute to goal setting include its relation to self-efficacy and its effect across students, time, tasks and other dependent variables. Goals are set in order to reach a specific performance outcome (Locke & Latham, 2006).

- *Active, achievement-oriented:* Achievement goals are used to motivate learning and are used in achievement settings. Active learning goals are used for active coping, long-lasting motivation and high achievement in face of challenges (Grant & Dweck, 2003).

- *Performance-oriented:* Performance goals enhance learning and are linked to learning-related outcomes. The focus of performance goals is to demonstrate one's competence, while the focus of mastery goals is to develop one's competence (Linnenbrink, 2005).

- *Intrinsic, internal:* Goals represent individual internal wishes and desire and are closely related to the need to achieve. As goals reflect individual motivation, implicit motives are closely related to explicit goals, which are then linked to individual performance (Pintrich, 2003).

Moreover, a programming education environment that considers the needs of the students is found to increase successful learning outcomes especially for novice students. Such an environment allowing collaboration and the exchange of quality information is found to reduce the rate of the dropouts in programming studies (Andersen et al. 2003; Xia, 2015).

Research design and methodology

Salinger et al. (2008) proposed that adopting qualitative research methods in studies of teaching programming is more conducive to the understanding of the given process than quantitative methods which might produce findings that lack detail in explaining the variation of experiences. In this study, qualitative research was especially useful in deriving meanings from objective conceptual descriptions of the programming experience. It was chosen because the aim was to capture individual student's experience of programming education. Lemke (2012) stated that, in educational research, the analysis and processing of verbal data can be done using content analysis in order to classify patterns, concepts and interactions. Further, Mayring (2000) suggested content analysis as the central method in analysing interview transcripts and it has been used in this study as a valid and effective method for making meaningful inferences from the given text data via stepwise analysis. It is also held that content analysis is appropriate where the research question is specific and that a step-by-step analysis is deemed appropriate, which is the case in this study.

Course data from introductory programming courses from the Aalto University in Finland was collected for a total of five years (2009-2013). The introductory programming course is offered twice per year, that is, in each of two academic semesters (Spring and Autumn/Fall). In 2013, the introductory programming course, which aimed to teach students the Python programming language, consisted of 4 hours of lecturers, 32 hours of self-learning, 77 hours of exercises and 20 hours reserved for the exam and exam preparation. The students, predominantly from first and second year, had access to printed materials, the course book and an online forum. In 2009, data was collected from 461 students; in 2010 from 390 students; in 2011 from 363 students; in 2012 from 229 students; and in 2013 from 212 students.

The aim was to collect data about these programming students including their background, motivation, choice of major and their experiences in programming. The data from these students has provided valuable feedback on the quality of teaching in programming education as well as the students' learning preferences. The students were given the chance to freely express themselves after the course was completed. These open responses were transcribed into a text and analysed in order to capture the variation in the students' experience in programming education in response to the research question, that is: *What are the different factors that affect the kind of learning goals students set for themselves in programming education*. The analysis adopted an iterative process and aimed to capture the pedagogical patterns evident in the verbal data. It is acknowledged that content analysis is context-dependent. This process also invites selection bias problems. Nevertheless, inductive content analysis is useful as a method for us to study in depth the underlying construction of the given phenomenon.

Results

Table 1 presents a classification of the different factors that affect the kind of learning goals that students set for themselves in programming education based on the students' experiences. It is important to note that the classification of learning goals in Table 1 is based on the feedback results and takes into account the student background and the direct quotes are only excerpts from the extended survey. The example excepts are intended to reveal the student profiles of our research and were translated from the original language from the course survey. Based on the research results, it is seen that learning goals are, in general, positively affected by interests in programming, background in computer science, personality traits, cognitive abilities and personal attitude towards the programming courses. In addition, we see that learning goals vary by nature and are dependent on the students' expectation and motivation towards programming. Moreover, learning goals can also be broken down into smaller more digestible forms at different stage of learning. For students with no prior background in computer science, programming is found to be a difficult skill to learn, and thus having well-defined learning goals is found to support learning.

Table 1. *Factors that affect learning goals and learning outcomes in programming education*

Aspects	Learning Goals	Example Student Excerpts
Prior Experience	Students with prior experience in computer science set internal goals to learn more.	• *In my opinion, a lecture should not waste time on independent learning. ... because advanced students with prior knowledge often have to wait for the beginners when exercises are solved at their own pace.*
Engineering Background	Students having an engineering background find the learning goals of programming to be encouraging.	• *The lecture notes from the course had all the necessary information I needed as well as plenty of good exercises. Because of this, I rarely needed to use other types sources for my learning.*
Personality	Students who are intrinsically driven set challenging goals and are interested in developing and extending their own learning.	• *The programming tool was amazing. It allowed effective distance learning and presented the exercises clearly. It was great that the tool also gave feedback on where I made mistakes and where to improve.*
Academic Background	Students with a strong academic background set learning goals based on a demand for both skills and grades.	• *The exam required plenty of information to be remembered by heart that was almost impossible. However, the course also taught a lot of new and useful knowledge to students who have never programmed before.*
Interests	Students with interests in computer science set realistic goals that motivate their studies.	• *The lectures were really clear and useful. If I had participated in all of the lectures, I would surely have learned everything.*
Cognitive Ability	Goal setting in programming may use cognitive abilities such as analytical thinking.	• *The last part of the programming exercises was extremely challenging. I had to use a tremendous amount of mathematical thinking and wrote over 100 lines of codes to solve it.*
Quality of Teaching	Active goals set by the lecturer aid learning. These goals can be set at different stages of learning.	• *The lecture was well presented and extremely clear even for me as a beginner in programming. The lecturer did not just assume but really took the needs of the students who have never programmed before into consideration.*
Student Expectation	Goals set by the students may differ from the goals set by the lecturer as students may have different expectations for the course.	• *I participated in the first few lectures but they felt quite tedious and did not help me very much. Learning by doing was a much better learning format for me especially when it comes to the exercises.*

Based on the results, the majority of students in our sample had prior experience with and interest in computers. The main reason why students chose computer science as their major is typically due to job demand and interest., A student's personality was found to contribute positively to learning outcomes. Having strong mathematical skills from previous studies and knowledge in related fields are conducive to learning programming as it requires high cognitive skills and personal interest coupled with an intrinsic interest in extending an individual's understanding.

Discussion and conclusion

Programming studies are the cornerstone of computer science. According to Carter and Jenkins (1999), one of the primary teaching goals of introductory programming courses is to equip students with basic knowledge and skills in programming. Lewis (2010) similarly confirmed that the goal is to develop students' programming skills and attitude towards programming. However, research results have shown that a large group of students do not attain the level of skills set by the teaching goals of the programming courses and fail to learn programming skills at an acceptable level (see, for example, Dehnadi & Bornat, 2006). Robins (2010) suggested that the reasons behind this phenomenon remain unexplained. The results in this study have shown that students with different backgrounds view programming differently and have different learning preferences. The high failure rate of programming courses might be improved by teachers' understanding of students' expectation and learning habits in these programming courses.

The present study investigated the factors that affect the kind of learning goals that students set for themselves. The main topics of this paper are students' goal setting and educators' intended learning outcomes in programming courses. The research question focused on one aspect, namely, factors affecting students' goal setting. Although research about teaching and learning programming has been researched in detail before, the focus of this paper concentrates on the pedagogic aspects of learning programming from the students' point of view via an empirical study, which might be considered as a new dimension. In other words, the paper potentially contains two interesting contributions. The first is the classification of aspects of learning goals based on a literature review; and, secondly, it presents findings on how different factors affect students' goal setting and learning. Instructors and researchers will benefit from understanding students' self-set goals so that they can deliver the appropriate content at a pace and in a form to match these personal goals.

This study thus confirms Whalley et al.'s (2006) finding that learning is more successful when students are able to understand, reflect and synthesise on a deeper cognitive level. Having well-defined learning goals appears to be closely related to successful learning outcomes and students' progress can be monitored and measured against the learning goals via regular feedback. In the present study, the concept of a learning goal is multifaceted and context-dependent. From the student's point of view, goal setting is dependent on factors such as prior knowledge on the subject, family background, expectation and motivation towards programming. When it comes to future research, it would be interesting to capture the students' perspective on the teaching methods and the quality of teaching in a personalised learning environment.

References

Anderson, L. W., Krathwohl, D. R., Airasian, P. W., Cruikshank, K. A., Mayer, R. E., Pintrich, P. R., Raths, J., & Wittrock, M. C. (2001). *A taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives.* New York: Longman.

Andersen, P. B., Bennedsen, J., Brandorff, S., Caspersen, M. E., & Mosegaard, J. (2003). Teaching programming to liberal arts students: a narrative media approach. *Proceeding, ITiCSE '03 Proceedings of the 8th Annual Conference on Innovation and Technology in Computer Science Education*, 109-113.

Biggs, J. B., & Collis, K. F. (1982). *Evaluating the quality of learning: The SOLO taxonomy.* New York: Academic Press.

Bloom, B. S., Engelhart, M. D., E. J., Hill, W. H., & Krathwohl, D. R. (1956). *Taxonomy of educational objectives: The classification of educational goals. Handbook I: Cognitive domain.* New York: David McKay.

Booth, S. A. (1992). *Learning to program: A phenomenographic perspective*. Unpublished PhD thesis, University of Gothenberg, Sweden.

Bruse, C., Buckingham, L., Hynd, J., McMahon, C., Roggenkamp, M., & Stoodley, I. (2004). Ways of experiencing the act of learning to program: A phenomenographic study of introductory programming students at university. *Journal of Information Technology Education, 3*, 143-157.

Buck, D., & Stucki, D. J. (2000). Design early considered harmful: Graduated exposure to complexity and structure based on levels of cognitive development. *SIGCSE Bulletin*, *32*(1), 75–79.

Carter, J., & Jenkins, T. (1999). Gender and programming: What's going on? *SIGCSE Bulletin*, *31*(3), 1–4.

Dehnadi, S., & Bornat, R. (2006). *The camel has two humps (working title)*. Retrieved from http://www.eis.mdx.ac.uk/research/PhDArea/saeed/paper1.pdf

Diamond, J. (1987). Soft sciences are often harder than hard sciences. *Discover,* 8, 34–39.

Eccles, J. S., & Wigfield, A. (2002). Motivational beliefs, values and goals. *Annual Review Psychology, 53*(109), 110-127.

Falout, J., Elwood, J., & Hood, M. (2009). Demotivation: Affective states and learning outcomes. *System, 37*, 403-417.

Hmelo-Silver, C. E., & Barrows, H. S. (2006). Goals and strategies of a problem-based learning facilitator. *Interdisciplinary Journal of Problem-Based Learning,* 1(1). doi:10.7771/1541-5015.1004

Jeffries, R., Turner, A. A., Polson, P. G., & Atwood, M. E. (1981). The processes involved in designing software. In J. R. Anderson (Ed.), *Cognitive Skills and Their Acquisition* (pp. 255–283). New York: Lawrence Erlbaum.

Johnson, S. D. Aragon, S. R., Shaik, N., & Palma-Rivas, N. (2000). Comparative analysis of learner satisfaction and learning outcomes in online and face-to-face learning environments. *Journal of Interactive Learning Research, 11*(1), 29-49.

Krathwohl, D. R. (2002). A revision of Bloom's taxonomy: An overview. *Theory into Practice*, *41*(4), 212–218.

Lemke, J. L. (2012). Analyzing verbal data: Principles, methods and problem. In K. Tobin & B. Fraser, (Eds.), *International Handbook of Science Education* (pp. 1175-1189). London: Kluwer Academic.

Lewis, C. M. (2010). How programming environment shapes perception, learning and goals: Logo vs. scratch. *Proceeding, SIGCSE '10 Proceedings* of the 41st ACM technical symposium on Computer science education, 346-350.

Lim, D. H., & Morris, M. L. (2009). Learner and instructional factors influencing learning outcomes within a blended learning environment. *Educational Technology & Society, 12*(4), 282-293.

Lister, R. (2004). Teaching Java first: Experiments with a PIGS-early pedagogy. In *Proceedings of the Sixth Australasian Conference on Computing Education*, ACE '04, 177–183. Australian Computer Society.

Lister, R., & Leaney, J. (2003). Introductory programming, criterion referencing, and Bloom. *SIGCSE Bulletin*, *35*(1), 143–147.

Locke, E. A., & Latham, G. P. (2006). New directions in goal-setting theory. *Current Directions in Psychological Science, 15*(3), 265-268.

Maher, A. (2004). Learning outcomes in higher education: Implications for curriculum design and student learning. *Journal of Hospitality, Leisure, Sport and Tourism Education, 3*(2), 46-54.

Mayring, P. (2000). Qualitative content analysis. *Forum Qualitative Sozialforschung / Forum: Qualitative Social Research*, *1*(2), Art. 20. Retrieved from http://www.qualitative-research.net/index.php/fqs/article/viewArticle/1089/2385

Nahrgang, J. D., DeRue, D. S., Hollenbeck, J. R., Spitzmuller, M., Jundt, D. K., & Ilgen, D. R. (2013). Goal setting in teams: The impact of learning and performance goals on process and performance. *Organizational Behavior and Human Decision Processes, 122*(1), 12-21.

Pintrich, P. R. (2003). A motivational science perspective on the role of student motivation in learning and teaching contexts. *Journal of Educational Psychology,* 95(4), 667-686.

Potter, B. N., & Johnston, C. G. (2006). The effect of interactive on-line learning systems on student learning outcomes in accounting. *Journal of Accounting Education, 24*, 16-34.

Rhee, B., Verma, R., Plaschka, G. R., & Kickul, J. R. (2007). Technology readiness, learning goals and eLearning: Searching for synergy. *Decision Sciences Journal of Innovative Education, 5*(1), 127-149.

Robins, A. (2010). Learning edge momentum: A new account of outcomes in CS1. *Computer Science Education*, *20*(1), 37–71.

Romanov, K., & Nevgi, A. (2006). Learning outcomes in medical informatics: Comparison of a WebCT course with ordinary web site learning material. *International Journal of Medical Informatics, 75*, 156-162.

Salinger, S., Plonka, L., & Prechelt, L. (2008). A coding scheme development methodology using grounded theory for qualitative analysis of pair programming. *Human Technology, 4*(1), 9-25.

Simon, B., & Taylor, J. (2009). What is the value of course-specific learning goals. *Journal of College Science Teaching, 39*(2), 52-57.

Valle, A., Nunez, J. C., Cabanach, R. G., Gonzalez-Pienda, J. A., Rodriguez, S., Rosario, P., Munoz-Cadavid, M. A., & Cerezo, R. (2009). Academic goals and learning quality in higher education students. *The Spanish Journal of Psychology, 12*(1), 96-105.

Whalley, J. L., Lister, R., Thompson, E., Clear, T., Robbins, P., Kumar, P. K. A., & Prasad, C. (2006). An Australasian study of reading and comprehension skills in novice programmers, Using the Bloom and SOLO Taxonomies. In *Proceedings of the 8th Australasian Conference on Computing Education,* ACE '06, pp. 243–252. Australian Computer Society.

Xia, B. S., & Liitiäinen, E. (2016). Implications of higher education on earnings: an empirical analysis of skills and competence. *International Journal of Adult Vocational Education and Technology, 7*(1), 50-64.

Xia, B. S. (2015). Benefit and cost analysis of massive open online courses: Pedagogical implications on higher education. *International Journal of Cyber Behavior, Psychology and Learning, 5*(3), 47-55.

Xia, B. S. (2013). Learning outcomes and knowledge sharing using web-based technologies in the Finnish forest education from an educational point of view. *E-Learning and Digital Media. 10*(1), 95-106.

Xia, B. S., & Liitiäinen, E. (2014). Economics of education and work life demand in terms of earnings and skills. *Citizenship, Social and Economics Education, 13*(1), 67-77.

Acknowledgement
The author is grateful for the key comments presented by Prof. Dr. Lauri Malmi and Dr. Päivi Kinnunen as well as their valuable contribution to this study. The author would like to thank Florilla Consulting Company for the funding of this project. The author is also grateful to the anonymous reviewers for their helpful and constructive comments.