Term Paper

Using the ARCS Model in Computer Programming Courses

Haisong Ye

CIMT 689

Indiana State University

April 12th, 2013

Abstract

Students in undergraduate educational technology program in China are required to take computer programming courses. The performances of these courses are not satisfied. Results from researchers show that difficulty of computer programming is not the only reason for the unsatisfied performance. Learners' beliefs, cognitive abilities, personalities, and motivations in learning play important roles in learning computer programming. This paper is to use ARCS motivational design model in the fundamental web programming course at a university in Shanghai. The target learners are junior undergraduate students who have already taken object-oriented programming, relational database design, and website design as prerequisite courses. This paper analyzes learners' motivational problems in the four categories of the ARCS model; defines motivational objectives and assessments; proposes motivational strategies at the module lever for the course. The strategies in this paper could be applied in other computer programming courses.

Introduction

Computer programming, as a fundamental skill in information technology, is recommended for undergraduate students in the major of educational technology in China. Educational Guidance Committee for Higher Institute of Educational Department of China (2010) recommends five course groups for different career goals, including *instructional system design*, *educational digital media*, *information technology education*, *educational software engineering*, and *educational equipment technology*. The five groups all have computer programming in their fundamental courses, and educational software engineering group has more programming courses, such as object-oriented programming, fundamental web programming, and multimedia programming in core courses (Educational Guidance Committee for Higher Institute of Educational Department of China, 2010). Students struggle to successfully complete computer programming courses, and the reasons are attributed to difficulty (Woszczynski, Guthrie, & Shade, 2005). Results in a survey of students of educational technology from a university in Shanghai show that students think computer programming is difficult and does not directly help them in their works (The Department of Educatioanl Technology of Shanghai Normal University, 2011). Based on motivational theories, difficulty might let students think that they don't have the ability to learn computer programming, and indirect help in job might lead to non-interesting or performance avoidance goal, which will affect students' motivation in learning computer programming (Mayer, 2008). This paper will address the motivation problems in computer programing courses by providing evidence in previous researches, and foster students' motivation by using the ARCS motivational design model.

Literature Review

The literature review focuses on four perspectives, motivation, ARCS model, researches in computer programming instruction, and motivation assessment.

Motivation

Schunk (1990) defined motivation as “the process whereby goal-directed behavior is instigated and sustained” (p. 3). Motivation is originated from curiosity and interests, goals and goal orientations, and self-efficacy beliefs (Driscoll, 2005). Interesting events, variety of instructional approach, fantasy, and creating a problem situation can stimulate curiosity and maintain attention in learners; by setting challenging, proximal, and learning goals lead better motivational learning behaviors; and, higher self-efficacy leads sufficient effort in learning and controls learning performance (Driscoll, 2005). For continuing motivation, satisfying expectancies and attributions are the possible methods. “Motivation appears to be enhanced when learners’ expectancies are satisfied and when they attribute their successes to their own efforts and effective learning strategies” (Driscoll, 2005, p. 331). Motivation plays an important role in instruction and learning.

ARCS Motivational Design Model

Keller (1999) concludes that motivation can be approached systematically; and learning environments “have strong impacts on both the direction and intensity of a person’s motivation” (p. 47). The ARCS model is based on four motivational conditions, gaining and sustaining attention, enhancing relevance, building confidence, and generating satisfaction (Keller, 1987; Keller & Suzuki, 2004). Keller and Suzuki (2004) expended the original seven-step systematic approach (Keller, 1997) to a ten-step process, which includes obtaining course information, obtaining audience information, analyzing audience, analyzing existing materials, listing objectives and assessments, listing potential tactics, selecting and designing tactics, integrating with instruction, selecting and developing materials, and evaluating and revising.

Although the ten-step approach is sophisticated, the approach is too complex and not easy for instructors who are not well trained. Suzuki (Suzuki and Keller, 1996) developed a simplified approach for motivational design in a project, which was in Japan with a team of twenty-five teachers in eight subject areas for developing computer-based instructions. This approach is represented in a matrix. The columns of the matrix are categories in ARCS model, which are attention, relevance, confidence, and satisfaction. The first row is learner characteristics to list overall motivation to learn. The second row is for learners' motivation of learning tasks. This row describes learners' interesting level, confidence, and their expectations for learning outcomes. The third row is about learners' attitude to instructional media. The fourth row is about learners' attitude to delivering method. Each of the items in the first fourth rows is marked with a plus or minus signs to indicate the positive and negative effect on motivation. The fifth row is for motivational strategies for each category in ARCS model based on the information in these first four rows. Suzuki's (Suzuki and Keller, 1996) research is computer-based instruction, so they have rows for medium and courseware. This simplified process was modified and used in a prototype of motivational adaptive computer-based instruction (CBI) and distance learning course offered by a university in United Kingdom (Keller, 1999).

Driscoll (2005) presents Keller's ARCS motivational design model in a four-step process. The first step is analyzing the audience. A learner's profile is recommended to identify any gaps in motivation. If learners are already motivated, or the problem is not about learner's motivation, it is not necessary to employ motivational strategies. The second step is to define motivational objectives. That is "what change in learner performance or attitude is to be expected from achievement of this goal?" (Driscoll, 2005, p. 342). The third step is design a motivational strategy, which is to select specific motivational strategies and integrated them into instruction. The fourth step is to try out and revise as necessary. Motivation, which needs more attention from the instructor for the effects of strategies, "should be thought separately from other aspects of instruction (Kefler, 1987b)" (Driscoll, 2005, p. 343).

Cognitive Profile and Self-Efficacy in Computer Programming Instruction

Woszczynski et al. (2005) conducted a study on the relationships of cognitive profiles and performance in a computer programming course, programming principles I (CS1). Woszczynski et al. (2005) used four distinct cognitive profiles in Krause's book *How We Learn and Why We Don't* (as cited in Woszczynski et al., 2005) that include Sensor Thinker (ST), Sensor Feeler (SF), Intuitive Thinker (NT), and Intuitive Feeler (NF) to classify their learners (as shown in table 1). Krause's cognitive profile inventory (CPI) was used as a classification method, which has "shown close relationships to problem solving ability" (Woszczynski et al., 2005, p. 295).

Table 1

Learners' characteristics in different cognitive profiles (adapted from Woszczynski et al., 2005, p. 295)

Type	Learners' Characteristics
ST	Prefer to study by themselves with little distraction
SF	Use structured thought processes and learn through repetition and breaking

	problems in to steps or milestones
NT	Tend to use pictures to enhance learning.
NF	Do well when they are allowed to build concepts from nothing and given freedom to try different ideas that may expand upon existing theories

The topics in CS1 cover standard programming constructs, data structures, algorithm, problem-solving strategies, machine representation, graphics, and networking (Woszczynski et al., 2005). Woszczynski et al. (2005) argued that “problem solving plays an important role in writing computer programs” (p. 295), so they believe that “the CPI provides appropriate personality test for measuring performance in CS1” (p. 295).

Woszczynski et al. (2005) found that learners in NT group achieved the highest averages, followed by NF group and ST group. The learners in SF group didn’t do very well. There is no evidence that gender affects performance in CS1. According to learners’ characteristics in SF group, instructional strategies like using web based and visual programing environment, building small programming modules, relating real life experience, and paired programming are suggested.

Hall, Cegielski, and Wade (2006) evaluated the relationship among computer programming task performance and a student’s theoretical value belief, cognitive ability, and personality. Hall et al. (2006) measured the theoretical value belief through the application of the Allport-Vernon-Lindzey (AVL) Study of Values; assessed individual cognitive ability by tests selected from the 1976 Kit of Factor-Referenced Cognitive Tests published by Educational Testing Services (as cited in Hall et al., 2006); and obtained personality through the aggregation of Rosenberg’s self-esteem, Judge, Locke, Durham, and Kluger's self-efficacy, Levenson's locus of control, and Eysenck & Eysenck's neuroticism measurement instruments (as cited in Hall et al., 2006). Learners' performances were assessed by the combination of written examinations, lab exercises, and individual comprehensive programming projects. The results show that performance is significantly positively correlated with theoretical value belief ($r = .225$, $p < .01$), personality ($r = .209$, $p < .05$), and cognitive ability ($r = .185$, $p < .05$). Theoretical value belief is significantly positively correlated with cognitive ability ($r = .181$, $p < .05$), but is not significantly negatively correlated with personality ($r = -.051$, $p > 0.5$).

Hall et al. (2006) suggested several ways to improve learning performance, which includes presenting logic before a discussion of syntax, describing esoteric programming concepts and syntax in analogies that students can understand, setting smaller sections in lab to lead more interaction, and developing students with both confidence and competence.

Motivation assessment

Mayer (2008) presents an simple version of instrument for motivation assessment (as shown in table 2) by integrating Pintrich and De Groot's Motivated Strategies for Learning Questionnaire (MSLQ); Roeser, Midgley, & Urdan's Patterns of Adaptive Learning Survey; and Borkowski, Weyhing, and Carr's measures of students' beliefs about learning (as cited in Mayer, 2008).

Table 2

Items of concerning your beliefs, feelings and expectations about a course (adapted from Mayer, 2008, p. 492).

Item	Scales	Measurement

	7	6	5	4	3	2	1	
I am interested in the material taught in this course.	○	○	○	○	○	○	○	Interest
I find the content of this course to be useful to me.	○	○	○	○	○	○	○	
I know that I will be able to learn the material for this class.	○	○	○	○	○	○	○	Self-efficacy
I am sure I can do an excellent job on the problem and tasks assigned for this class.	○	○	○	○	○	○	○	
If I perform poorly on a test in this class, it is because I did not try hard enough to learn the material.	○	○	○	○	○	○	○	Effort-based attributions
Doing well in this class depends on how much effort I give.	○	○	○	○	○	○	○	
In this class, I like problems and materials the best that really make me think.	○	○	○	○	○	○	○	Mastery achievement goals
I like problems and tasks that I can learn from in this class, even if I make a lot of mistakes.	○	○	○	○	○	○	○	

For each item, 1 indicates that the statement is not true and 7 indicate that the statement is true. The first two items measure interest, third and fourth items are for self-efficacy, fifth and sixth items are for effort-based attributions, and the last two items are for mastery achievement goals. The sum of two items evaluates learners' level on each measurement, for example, if the sum of the first two items is greater than 10, it means the learner is interested in this course (Mayer, 2008).

Keller (2010) introduces the course interest survey (CIS) instrument to measure motivational level for course. CIS has 34 items for four ARCS categories with scale range from 1 to 5. Evidence shows that CIS has internal consistency (Keller, 2010).

Another instrument suggested by Keller (2010) is Instructional Materials Motivation Scale (IMMS). IMMS has 36 items cover four ARCS categories.

Summary

Several studies (Hall et al., 2006; Woszczynski et al., 2005) indicate that difficulty is not the only reason for unsatisfied performance in computer programming courses. Personality, especially for learning style or cognitive profile, and self-efficacy can be important factors. Therefore foster learning motivation for students with different cognitive profile and at different self-efficacy can be a possible way to improve their learning performance.

Keller's ARCS motivational design model (Keller, 1987, 1997, 1999; Keller & Suzuki, 2004) is a well-designed systematic approach for motivational design. The four-step process which is presented by Driscoll (2005) and the matrix of motivational design which is proposed by Suzuki and Keller (1996) simplify the process of ARCS model. This paper will use the four-step approach to design motivational strategies in computer programming courses.

Mayer (2008) suggests a simple measurement instrument that can used as a quick evaluation of learners motivation; and Keller (2010) suggests two sophisticated instruments for motivational measurement. Both could be adapted and used in my design.

A Motivational Design Case: Fundamental Web Programming

Course Overview

Fundamental Web Programming (FWP) is a required course for undergraduate students in educational technology at a normal university in Shanghai, China. FWP is an introduction course for server-side data-driven technology. We use ASP.NET as the main server-side technology, C# as the default programming language, and Microsoft SQL Server as default database server. Students who finish this course are expected to be able to build a small size ASP.NET website with C#. Small size means that there are three to four tables in database, except the tables for website security (authentication, authorization, and security logs), because these tables are pre-defined within ASP.NET security mechanism. The content of FWP covers the standards in guidelines of educational technology curriculum, section CC0405 (Educational Guidance Committee for Higher Institute of Educational Department of China, 2010).The prerequisite courses of FWP are *C# programming (object-oriented programming)*, *Relational Database Design and Application*, and *Website Design*.

Learner Analysis for Motivational Problems

The target learners are junior undergraduate students in the major of educational technology. They all have taken prerequisite courses before they take FWP. The motivational problems are analyzed from four categories of ARCS.

Attention. Initial attention is variable. This is a required course, so not all students are interested to this course. Based on my observation in previous semesters, around 20% students are interested and want to learn the skills in web programming. These students' initial attention is high. About 40% students have moderate initial attention, they are not sure what will they learn in this course, but they still have some level of curiosity in this course. The rest 40% students are not interested in this course. There are two reasons. One is from their experience in previous programming courses; the other is that they cannot see the value of this course in their future study or career.

Relevance. Relevance is variable. Students in this class can be in three academic concentrations of educational technology, instructional design, educational digital multimedia, and educational software engineering. This course is highly relevant to students in educational software engineering, but not for students who concentrate in instructional design and educational digital multimedia.

Confidence. Students’ confidence is low to moderate. Most students didn’t do well in prerequisite courses, which are C# programming and relational database design, so they have low confidence to succeed in this class. Some students think this course is more difficult, because this course integrates knowledge and skills of programing, database, and web design.

Satisfaction. Satisfaction is variable. As stated in the attention section, most students do not expect the outcome of this course will be helpful for their future study and career. Another aspect is that as an undergraduate level course, the final grade is based on percentages of the number of students, which means that only 20% students or less can get A, only 40% students or less can get B, and no more than 60% students can get grade higher than B. Some students in the past did well in the class, but they didn’t get appropriate grades for their work, because of the limitations of grade percentage.

Motivational Objectives and Assessments

Since FWP is a face-to-face course, it possible to observe learners’ motivational behavior while taking the course. Besides observation, learners will fill module-level questionnaires periodically for their motivational indicators. Table 3 is a summary of motivational objectives and assessments for this course.

Table 3

Motivational objectives and assessments

	Motivational design objectives	**Assessments**
Attention	• Learners will be stimulated and gain curiosity of the course. • Learners will keep their attention level through the whole course.	• The instructor will observe the amount and frequency of learners' distractive behaviors in classroom. • Learners will be asked about the interestingness of the course through informal personal talks and periodically questionnaires.
Relevance	• Learners will find that the content of the course are relevant to their career concentrations and how they could apply it in real situations.	• Learners will be asked about how they feel the course can help them in real situations through informal personal talks.
Confidence	• Learners will get confidence after completing the first four lessons. • Learners will have satisfactory levels of confidence during the course.	• The instructor will observe whether learners can complete the practice exercises in classroom within expected amount of time (various in 40~60 minutes). • Can learners complete the after class assignments on time. • Learners will fill out a

		questionnaire to check their confidence level. • The instructor will have small talks with students to check their confidence level.
Satisfaction	• Learners will be able to identify that the outcome of this course are benefit to their future study and career. • Learners will satisfied with what they learned other than the final grades.	After completion of the course, • Learners will be asked to list some possible applications of data-driven web sites in their future study and career; • Learners will fill out a questionnaire for their satisfaction level for the whole course.

Motivational Strategies

The FWP contains four modules, introduction, using web forms and controls, interacting with database, creating secure website, and building your website. To gain target learner's attention and build their confidence, a simple but complete example will be presented in module one. This example will also be their first output after the first module. Through this walk through project, learners would know what they can do. In the second, third, and fourth modules, tasks are designed holistically. Learners will be initialized by a series of problems in the simple project the finished in module 1, then learners will learn from examples and apply the skills into the project to improve the user interface, data representation, and security. During these modules, instructor will monitor learners' motivation status by different approaches, including individual discussion, questionnaires, and observation. Finally, learners will build their own website in group. The instructor will provide advices on the size of the project. Small size project is not challenge and large size project might not feasible at this level. Inappropriate project size could affect learners satisfactory and confidence. Table 4 presents motivational strategies for FWP at the module level.

Table 4

Motivational Strategies for FWP

Modules	**Motivational strategies**
Introduction (6 hours)	**Beginning** • Present profile icons of popular social websites and let learners identify which websites they are. (A, R) • Let learners use a simple version of twitter in class, and tell them that this is their first walkthrough project. (A, R) • State that the longest code block is only 5 lines and no manual script for database query is needed. (C) • State that there will be a walkthrough screen tutorial for learners to follow. (C) **During** • Ask learners to summarize what characteristics of the simple twitter. (R, S) • Divide learners into groups of 2 to develop the walkthrough

	project. (C) • Provide help when learners ask. When help leaners, try to guide them instead of give them a solution directly. (C, S) • Encourage 2-3 groups to present their projects. (A, C, S) • Provide positive feedback. (C, S) **End** • Provide a summary of objectives the learner completed in this module. (C, S) • Provide leading questions for next module. (A)
Using Web Forms and Controls (12 hours)	**Beginning** • Review the simple twitter project. (A) • Discuss where the user interface/user interaction of the simple twitter project could be improved. (A, R) • State the learning objectives which can address the issues in previous activity. (A, R) **During** • Provide exercise tasks focus on single controls. These tasks would be connected with real situation, but simple. (R, C) • Post a problem, which is related to one or multiple web controls, and let learners discover the appropriate controls, and then try them on web form. (R, C) • Provide examples of web controls in real websites. (A, R, C) • Give positive feedback, when learners chose improper controls.

	(C) **End** • Provide a summary of objectives the learner completed in this module. (C, S) • Provide leading questions for next module. (A)
Interacting with Database (18 hours)	**Beginning** • Recall database interaction part of the simple twitter project. (A, R) • State that minimum database query script is needed. (C) • Present more sophisticated database interaction pages, and state these pages are using data controls without manual SQL query scripting. (C, S) • State the learning objectives with examples. (A, R, C) **During** • Provide progressive exercises from simple to complex interactions. (C, S) • Provide real examples in different context, especially related to instructional design, digital multimedia, and educational software. (R, S) • Encourage 2-3 learners to present their assignment. (A, C, S) • Give positive feedback and provide relative resources when learner showed lacking prior knowledge. (C) **End**

	• Provide a summary of objectives the learner completed in this module. (C, S) • Provide leading questions for next module. (A)
Creating Secure Websites (6 hours)	**Beginning** • Show examples of non-secured website. (A, R) • Present several typical password hack tricks and how these tricks work on ill-designed security mechanisms, e.g. sql inject. (A, R) • State the learning objective, which is to use ASP.NET website security in the simple twitter project. (A, R) • State that using ASP.NET website security needs few coding works. (C) **During** • Provide exercises that address different security situations. (R, C, S) • Let learners try to hack other leaners' website with the tricks presented at the beginning of class. **End** • Provide a summary of objectives the learner completed in this module. (C, S) • Provide leading questions for next module. (A)
Building Your Website (12 hours)	**Beginning** • Show examples of previous projects along with explanation of

group project criteria. (A, C)

During

- Provide advice on the size of projects. (C, S)
- Provide help as necessary. When providing help, using prior examples and exercises instead of giving learners solutions directly. (R, C)

End

- Present group projects and provide positive feedback. (C, S)
- Let learners enumerate possible application in their future study or career. (A, C, S)

Summary

The FWP course is a required course for undergraduate students, so not every student is interested in this course. Students think FWP is difficult, because it needs knowledge in programming and database. In addition, students cannot relate their future study and work with this course. Motivational problems exist. Based on learners' characteristic analysis and learners' attitudes to learning tasks, multiple motivational strategies are employed to foster learners' motivation in learning web programming. This design is a module level design based on the author's personal experience in teaching computer programming and doesn't have chance to tryout. A more detailed lesson level design should be developed in the future for real instruction.

Conclusion

Researchers find that motivation is a key element in successful instruction. Learners' motivation could be related with their interesting, learning goal, self-efficacy, and attribution. Keller's (1997) ARCS model provides a systematic approach to design strategies in improving learners' motivation through gaining attention, enhancing relevance, building confidence, and generating satisfaction.

This paper applies motivational strategies in the course of fundamental web programming for the undergraduate students in educational technology. Difficulty, lower relevance to future study and work, strict grading system, and lower curiosity level affect learners' motivation. Through the four-step approach, this paper analyzed the learners' characteristic and the learning context, and proposed several strategies to improve leaners' motivation in the course. The strategies in this paper might be applied to computer programming courses in other majors.

References

Driscoll, M. P. (2005). *Psychology of learning for instruction* (3rd ed.). Boston, MA: Pearson Allyn and Bacon.

Educational Guidance Committee for Higher Institute of Educational Department of China. (2010). *Standards of Educational Quality for the Major of Educational Technology in Higher Institute*.

Hall, D. J., Cegielski, C. G., & Wade, J. N. (2006). Theoretical Value Belief, Cognitive Ability, and Personality as Predictors of Student Performance in Object-Oriented Programming Environments. *Decision Sciences Journal of Innovative Education, 4*(2), 237-257.

Keller, J. M. (1987). The systematic process of motivational design. *Performance & Instruction, 26*(8), 1-7.

Keller, J. M. (1997). Motivational design and multimedia: Beyond the novelty effect. *Strategic Human Resource Development Review, 1*(1), 188-203.

Keller, J. M. (1999). Using the ARCS Motivational Process in Computer-Based Instruction and Distance Education. *New Directions for Teaching & Learning*(78), 39.

Keller, J. M. (2008). First principles of motivation to learn and e3-learning. *Distance Education, 29*(2), 175-185.

Keller, J. M. (2010). *Motivational Design for Learning and Performance: The ARCS model approach*. New York, NY: Springer.

Keller, J. M., & Suzuki, K. (2004). Learner motivation and E-learning design: a multinationally validated process. *Journal of Educational Media, 29*(3), 229-239.

Mayer, R. E. (2008). *Learning and Instruction* (2nd ed.). Upper Saddle River, NJ: Person Merrill Prentice Hall.

Schunk, D. H. (1990). Introduction to the special section on motivation and efficacy. *Journal of Educational Psychology, 82*, 3-6.

The Department of Educatioanl Technology of Shanghai Normal University. (2011). *A survey of graduated students of Shanghai Normal Univesrity in educational technology*

Woszczynski, A. B., Guthrie, T. C., & Shade, S. (2005). Personality and Programming. *Journal of Information Systems Education, 16*(3), 293-299.