Journal of Engineering Education Transformations ,
Volume 36 , No. 2 , October 2022 , ISSN 2349-2473, eISSN 2394-1707

10.16920/jeet/2022/v36i2/22151

The Complexities of Teaching a Program to Find the Largest of Three Numbers in the CS1 Course

Srinivasan Lakshminaryanan
Department of Computer Sciece and Engieering, Jain University, Karnataka.
srinrad@gmail.com

Abstract : This study explores social, cognitive, metacognitive, and affective aspects of teaching a simple programming problem by a lab instructor. Every programming problem in practice creates a unique teaching experience. The lab instructor is the implementer of the educational policies, curriculum, course design, institutional culture and represents the system for the student. As an essential point of contact between the system and the student, the instructor has an enormous opportunity to contribute to the students' learning. Therefore, there is a need to capture these experiences to improve the quality of instruction. A phenomenological lived experience method is adopted to describe teaching the example problem to find the largest of three numbers in a CS1 course. In the social domain, we present the difficulties faced due to differences in competency between instructors and teachers. In the cognitive domain, we present the difficulties due to the variations possible in the program and lack of time to teach the problem. In the metacognitive domain, we present the difficulties of engaging students at higher cognitive levels of applying, analyzing, and evaluating. Finally, in the affective domain, we present the difficulties related to acceptance, judgment, the time required for the relationship, and the need for completion.

Keywords : Critical Pedagogy, CS1, Introductory Programming Course, Lived Experience, Reflective Practice.

1. Introduction

Lab instructor (LI) is the teacher who interacts with the students while working on a program in programming labs. A lab instructor faces many difficulties that are different from those faced during lectures. Every teacher who teaches programming will encounter the program to find the largest of three numbers. Hence documenting the difficulties in teaching a common problem will start an emancipatory discourse about the unique difficulties the lab instructors face. The lab instructor (LI) has taught the problem many times over four semesters as a lab instructor in the CS1 course. The problem is complex since it can be implemented in many ways. Many reference implementations in textbooks and online resources are wrong. There are many ways this problem can be used. It can be used as a demonstration or exercise for the concepts of if statements, ternary operators, logical operators, and function. The learning theories applicable are constructivism, social constructivism, guided discovery, and didactic methods. Teachers can organize communication through peer learning, group discussion, lecture, synchronous discussion, and asynchronous

Srinivasan Lakshminaryanan
Department of Computer Sciece and Engieering,
Jain University, Karnataka.
srinrad@gmail.com

discussion. The teacher may use automated testing tools, tutoring tools, scaffolding tools, debugging, and tracing tools. The choices are endless. The choices get implemented on the days of teaching this problem. LI feels that one such story in different domains needs to be recorded. The domains are classified as social, cognitive, metacognitive, and affective domains.

2. Literature Survey

This work is inspired by Isomöttönen (2018) work and the relief the LI felt reading the article, which led to an exploration of the method of documenting lived experience and the emancipatory nature of the critical pedagogy. There is no freedom for a lab instructor in designing, implementing, and assessing a course in the Indian education setting. De George (2003) writes, "But academic freedom is necessary for a university dedicated to the pursuit of knowledge in a democratic society." An individual cannot change the system. Hence where freedom is not a given, the instructor will have to find that freedom in the existing circumstances. George (2010) mentions the ethical issues the instructors face while doing remedial teaching, related to the job security of the instructor based on student admissions and the need to inform the student's unsuitability for the course. Every year LI meets students who do not fit the engineering education but cannot openly discuss their difficulties with the students, potentially leading to change in their educational perspective.

Giroux (2003) states, "As neo-liberal capitalism substitutes market relations for the rule of justice and law, it becomes more difficult for educators, students, and citizens to address pressing social and moral issues in systemic and political terms." In India, educational institutions are not registered as businesses. However, economic viability plays a significant role in private institutions, resulting in fundamental ethical issues for the teachers. Being the lowest in the power hierarchy, the lab instructors cannot express opinions. However, the lab instructor is the only connection point between the system and the student, and they have to represent and defend the system even when there is a fundamental difference of opinion concerning education as a business and education as an emancipatory process.

Further literature survey reveals other issues, but very little published literature exists about the issues lab instructors face in Indian educational systems. All the issues in the systems finally manifest in the lab instruction. This paper gives a voice to those issues as a lab instructor's lived experience, who, without waiting for a systemic change, looks at what is possible to remedy the situation within the constraints.

3. Method

O'Hara (2018) defines "Autoethnographic writing is a scientific method which contextualizes experiences in cultural, social, political and personal history." Their work proposes a six-step process to document lived experiences.

Step 1. Identify whether the work is analytic or evocative. In this study, the work was performed by a single researcher, and the work is classified as analytic.

Step 2. Ethical responsibility. Since the study anonymizes the experiences and the domain is education, the study can be assumed to be ethically responsible.

Step 3. Theoretical underpinnings. Bloom's taxonomy, Metacognitive Awareness Inventory, Critical Pedagogy were studied to provide a theoretical basis to the study.

Step 4. Gather the data. Lab manuals, student submissions, websites, student feedbacks were collected.

Step 5. The collected data were classified under four main themes of the social, cognitive, metacognitive, and affective domains. The related difficulties are presented.

Step 6. Complete the report with collected data. The feedback from the students and anecdotal experiences are presented.

3. Social Domain

In the institution where LI works, Thareja (2012) is used as the textbook. The book lists the code for finding the largest of three numbers, as shown below:

```
if ( num1 > num2 && num1 > num3)

   printf("\n %d is the largest number", num1);

if ( num2 > num1 && num2 > num3)

   printf("\n %d is the largest number", num2);
```

```
else

  printf("\n %d is the largest number", num3);
```

The code does not work for inputs 2, 2, and 1. It prints 1 as the largest number. A later edition of the book may have fixed this problem, but the course syllabus does not reflect it. One of the instructors taught the wrong code in the class. LI had to mentor a subset of the students belonging to that class in the lab. LI found it uncomfortable to inform the instructor about the mistake because of his inhibitions and previous experiences of doing it. The previous experience resulted in a bad relationship and a permanent difficulty in the working relationship. This problem becomes even more complex when the instructor who does not know the code evaluates the code in an examination. Furthermore, as a lab instructor, one needs to ensure the relationship between the classroom instructor and the student remains intact. LI took a diplomatic route of giving the test case and asking the students to fix the problem in the lab. Students then needed to be counseled about the fallibility of teachers and how they can sometimes be wrong.

The organization does not have a mechanism for LI to report this issue. The instructors share knowledge informally in groups where they feel secure. An educational researcher is treated with suspicion. LI had to communicate that the purpose of the research is for the improvement of the course and not to expose individuals. It is a delicate matter, and no institution can afford to teach the program to find the largest of three numbers wrong. There is also this problem of who will bell the cat and inform the management. So even when the solutions are known, the problems do not get solved because of social constraints. It is impossible to fix all the wrong code out there on the internet and the books. Analysis of code submissions shows that every instructor had students who had submitted the wrong code. It is unclear how this situation will change since this must have been the case for many years now. Every year this institution is audited by various committees. It is surprising that this one problem escapes the scrutiny of all the eyes and manages to survive as if it has a life of its own.

4. Cognitive Domain

A survey of first-year programming lab instructors was conducted using semi-structured interviews and convenience sampling to understand the instruction design for teaching the problem of finding the largest of three numbers. Lab instructors covered more than one version of the code. They used the program to demonstrate nested if and logical operators. However, none taught more than three ways the problem could be solved. It is not clear how much time a student would take if they were to discover all the important variations of the programs by themselves. Even in a guided discovery mode with synchronous communication, the time required may still be more than what is provisioned for this problem in the course. Even in lecture-based approaches, instructors do not want to invest too much time on one single problem and cover all the important variations.

(Programiz (n.d.), Sanfoundry (n.d.), Beginnersbook (n.d.), w3resource (n.d.), Programming Simplified (n.d.), GeeksforGeeks (n.d.)) sites on google search results have programs that are not correct or are inefficient. This list above shows that none of the websites comprehensively deal with the problem. Most of the students' code comes from class notes, textbooks, or websites. Most of them are wrong or not efficient.

Table 1 elaborates on the sequence of steps required to understand many aspects of conditional statements and logical operators. The question LI faces is whether he should take the students through this elaborate flow of steps. There are many issues to be considered here. First, the time required is more than the time required, just one variation. There is no guarantee that a student will complete all the steps in the given time. The student may not be willing to invest additional time. If the feedback process is not synchronous, it may take days before the student finishes the program. It is not immediately apparent to LI whether it is necessary or practical for a novice programming student to explore all the listed variations of the program. LI took a simple position. He made himself available in the lab for additional hours. It was left for the students to learn as much as they wanted to know. The extreme apprenticeship method described in Vihavainen et al. (2011) was adopted to facilitate the students to complete all the listed steps

5. Metacognitive Domain

LI referred to Schraw and Dennison (1994) to understand the components of metacognitive awareness. Metacognitive awareness consists of metacognitive knowledge: declarative, procedural, and conditional; and metacognitive strategies: planning, monitoring, information management, debugging, evaluation. In addition, LI read the works of Volet and Lund (1994), Loksa et al. (2016) to understand metacognitive instruction in the context of teaching programming.

While teaching the largest of three numbers, when a student submits the program, the first time, metacognitive discussion happens about the program's indentation. Many students have difficulty indenting this program. Though there are tools available to indent the program, the entire focus is on making the program work. LI played the role of a customer and refused to read the programs that were not indented. This position taken by LI forces the student to apply the metacognitive strategy of organizing information. When the program fails for 2 2 1 input, the strategy of "read, understand and reproduce" code does not work. Now the student is forced to think and analyze where they went wrong. The metacognitive debugging happens, and students have to arrive at the strategy of tracing the program for the given input. When they trace that using paper and pencil methods, the students will use their mental model of execution, which might not be correct, and the program might work differently on the computer. They must use trace prints to understand the control flow and values of variables. The method of using trace prints becomes cumbersome because of the multiple trace prints they have to write. Then the next option is to learn debugging using a debugger tool. LI can wait for the student to recognize the need for this new method or give the method before the need for the method arises. LI can continuously watch the student and his level of engagement. When the student says, "I do not know what to do", LI can demonstrate the use of the debugger. LI invites all the students whenever he demonstrates debugger, yet he observed that nobody uses it. LI came up with a new saying, "if you do not use the debugger, you don't know programming," making it a part of the course's value system. The debugger does not solve the problem. It can only show where the control flow is wrong or which step is wrong. So, the students must come up with a hypothesis. Based on the hypothesis, a solution needs to be proposed and tested. The process of coming up with a hypothesis and experimenting are new strategies needed in higher-order cognitive processes of applying, analyzing, evaluating, and creating the taxonomy defined in Krathwohl et al. (2009).

Table 1 : Flow of Communication for Learning Important Variations of The Program

Section	Details

Step1:
if (a>b && a>c)
largest = a;
else if (b>a && b>c)
largest = b;
else if (c>a && c>b)
largest = c
What will happen if a is 2, b is 2 and c is 1?
	Step 6:
	if(a>b & a>=c)
	largest = a;
	else if (b>c)
	largest = b;
	else
	largest = c;
	
	Why did the previous solution work?
	Can you do without & or &&?

Step 2:
if(a>=b && a>=c)
largest = a;
else if (b>=a && b>=c)
largest = b;
else if (c>=a && c>=b)
largest = c;
Why did the solution work?
What will happen if you replace
else if (c>=a && c>=b) by else?
	Step 7:
	if (a> b)
	if (a>c)
	largest = a;
	else
	largest = c;
	else if b > c
	largest = b;
	else
	largest = c;
	Can you do without else statement?

Step 3:
if(a>=b && a>=c)
largest = a;
else if (b>=a && b>=c)
largest = b;
else
largest = c;
Why did the previous solution work?
Do you need b>=a?
	Step 8:
	largest = a;
	if (largest>b)
	largest=b;
	if (largest >c)
	largest =c
	Can you do without using if statement?

Step 4
if(a>=b && a>=c)
largest = a;
else if (b>=c)
largest = b;
else
largest = c;
Why did the previous solution work. Now what will happen when you replace
all >= with >, will it work?
	Step 9:
	largest =
	a>b?a>c?a:c:b>c?b:c;
	
	Can you do it using macro. Hint.
	#define max(a,b)
	(a>b?a:b)?

Step 5:
if(a>b && a>c)
largest = a;
else if (b>c)
largest = b;
else
largest = c;
	Step 10:
	largest =
	max(max(a,b),c);
	
	What will happen if you
	#define max(a,b)
	a>b?a:b?
	Caps

LI sees an enormous opportunity to engage with students in this problem in the metacognitive domain. However, LI could only engage with a small number of people. Most students do not want to go beyond the levels of reading, understanding, and reproducing programs.

LI believes that freedom is essential to learning. It is unclear what LI should do if a student does not want to learn beyond a certain point. The system does not demand it. A new program had to be written every lab session, and the class instructor assigned new exercises. The system moves on, leaving behind incomplete learning. LI constantly reminded the students about the effort they needed to invest in becoming a programmer. Beyond that, there was very little that LI could do. He had to get back to the drawing board to understand how to survive the system and still teach programming.

6. Affective Domain

When a student gets assigned to the LI, the story has already spread about the LI, and students have preconceived notions. The program to find the largest of three numbers is an example of a teacher and student interaction. Some students felt positive about the experience, some negative, and most were apathetic. Three positive feedbacks are presented in this article. In the conditions of complete freedom of choice, what happened is of significance since the student can always terminate the relationship without any consequences to grades except lost time. One student wrote in her feedback. "Actually, all my friends of last semester told me that let anyone be your lab in charge but not that LI sir, he will show you, people, the real hell. I was wondering why these people tell like that about that, sir. Later I came to know that how well we should prepare for his class. If we have lab tomorrow, we used to sit today and think about what he might tell us to experiment tomorrow in his lab. In my opinion, initially, I felt why this sir is so strict and torture us like anything. But as the days passed, I understood the real agenda behind the strictness showed by sir, like when we compare with others, I was able to do something that others suffer to do that. Hence, I want to thank him because I had "ZERO" knowledge about c programming, but now I know something about c. Last, I want to conclude it by saying, sir, please continue giving torture to your students and make them learn something which they don't know. we miss you, sir, with lots of love and bye, sir." LI always wondered why students feel tortured. They have paid to learn, and it appears paradoxical that they should hate the person who is committed to their learning. This anger is even more confusing because none of the learning is forced. It is optional. The problem is that the students are never sure whether it is truly optional, and even when it is truly optional, they want the instructor's approval. Interestingly, the other aspect is that when there is freedom, and the teacher is not willing to accept the lower quality work even though the grades are given according to the norm, students consider the teacher as strict.

Another student wrote, "I think learning CCP would have been quite easier, but LI sir made it a little difficult (No offense, just speaking frankly as I usually do). As we have reached the end of the semester, I am thankful to him for making me work hard on programs, for always asking me the same question repeatedly (Do you really want to learn programming? If yes, then why do I not see you working on programs as you promised?). This question is annoying sometimes, but this is the very question that made me explore more of c-programming. Last but not least, Thank you, sir!! for helping us to solve the programs by ourselves and always being available to clear our doubts." LI feels that two important practices followed were asking students whether they wanted to learn programming and helping them achieve it. Many research publications have addressed difficulties in programming. The process of the LI questioning students whether they want to learn programming, and being available when they are working, leads to learning for some students. The problem with this solution lies in instructor availability and student effort. Instructor availability is not scalable. Instructor time and student time increase when the learning quality has to be improved; hence, it might be of no interest to the educational systems, intending to scale up the educational business.

The following is feedback from a student taking the course for the third time after failing twice. "Feedback: It was fun and exciting to learn coding from my teacher. He was always helping me to fix the bugs and errors, helping me with the code. He always guided me with the best way to obtain output and made me aware of how to deal with the subject." LI made him repeat the programs until he could remember and understand the programs, which was sufficient for him to pass. He understood the importance of spaced repetition. He still has a long way to go before he can write programs. LI did not even attempt to teach him the program to find the largest of three numbers. One method of instruction does not fit all students. It is dependent on the capabilities and circumstances of each student. In the first student's case, planning and reflection were emphasized. In the second, focusing and experimenting strategies were used, and spaced repetition was used in the third student's case. LI wonders how one could know what instruction method would work for a specific student, given 60 plus learning theories, hundreds of instructional methods, and thousands of journal articles, and many prescriptions and opinions on the internet.

7. Conclusions

This lived experience is that of one individual instructor in a CS1 lab in the context of one single problem. This problem is deceptively simple computationally, yet it has the entire gamut of issues a lab instructor faces. Just making the problem a demo problem or ending the instruction after students' one or two successful attempts to solve the problem is an easy way out. LI took the path of going beyond the easy, pragmatic, or probably even the right way. Unless we take the uncharted path and document the experiences, new pathways may not be found. In the MOOCs and technology-enabled learning era, LI was surprised that he needed to help the student with what was sometimes right in front of their eyes. Looking at the body of knowledge from the perspective of teaching a single problem in a specific instructional setting helped the instructor to manage the amount of information available on the internet. Documenting lived experiences improved the process of continuous refinements of instruction and helped more students. The lived experiences can be helpful in the induction training of new teachers. It can also be used to compare the methods used by the teachers in different instructional settings and different cultures

References

[1] Axline, V. M. (1969). Play therapy. Ballantine Books.

[2] Beginnersbook. (n.d.). https://beginnersbook.com/2014/06/c-program-to-find-greatest-of-three-numbers((Accessed on 07/20/2020))

[3] De George, R.T. Ethics, Academic Freedom and Academic Tenure. Journal of Academic Ethics 1, 11–25 (2003). https://doi.org/10.1023/A:1025421706331

[4] Fortune-Wood, Jan. Without Boundaries Consent-Based, Non-Coercive Parenting and Autonomous Education. (2000).

[5] Geeksforgeeks. (n.d.).https://www.geeksforgeeks.org/c-program-to-find-the-largest-number-among-three ((Accessed on 07/20/2020))

[6] George, M. (2010). Ethics and motivation in remedial mathematics education. Community College Review, 38(1), 82-92. https://doi.org/10.1177/0091552110373385

[7] Giroux, H. A. (2003). Selling out higher education. Policy futures in education, 1(1), 179-200. https://doi.org/10.2304/pfie.2003.1.1.6

[8] Isomöttönen, V. (2018). For the oppressed teacher: stay real! Teaching in Higher Education, 23(7), 869-884. https://doi.org/10.1080/13562517.2018.1437131

[9] Krathwohl, D. R., & Anderson, L. W. (2009). A taxonomy for learning, teaching, and assessing: A revision of bloom's taxonomy of educational objectives. Longman. https://doi.org/10.1080/00461520903433562

[10] Loksa, D., Ko, A. J., Jernigan, W., Oleson, A., Mendez, C. J., & Burnett, M. M. (2016). Programming, problem solving, and self-awareness: effects of explicit guidance. In Proceedings of the 2016 chi conference on human factors in computing systems (pp. 1449–1461). https://doi.org/10.1145/2858036.2858252

[11] Programiz. (n.d.). Programiz. https://www.programiz.com/c-programming/examples/largest-number-three. ((Accessed on 07/20/2020))

[12] Programming simplified. (n.d.). https://www.programmingsimplified.com/c/program/largest-of-three-numbers((Accessed on 07/20/2020))

[13] Sanfoundry. (n.d.). Sanfoundry. https://www.sanfoundry.com/c-program-biggest-3-numbers/. ((Accessed on 07/20/2020))

[14] Schraw, G., & Dennison, R. S. (1994). Assessing metacognitive awareness. Contemporary educational psychology, 19 (4), 460 -475. https://doi.org/10.1006/ceps.1994.1033

[15] O'Hara, S. (2018). Autoethnography: The Science of Writing Your Lived Experience. HERD: Health Environments Research & Design Journal, 11(4), 14–17. https://doi.org/10.1177/1937586718801425

[16] Thareja, R. (2012). Computer fundamentals & programming in c. Oxford University Press.

[17] Vihavainen, A., Paksula, M., & Luukkainen, M. (2011). Extreme apprenticeship method in teaching programming for beginners. In Proceedings of the 42nd ACM technical symposium on computer science education (pp. 93-98). https://doi.org/10.1145/1953163.1953196

[18] Volet, S., & Lund, C. (1994). Metacognitive instruction in introductory computer programming: A better explanatory construct for performance than traditional factors. Journal of educational computing research, 10 (4), 297–328. https://doi.org/10.2190/9A08-Y2Q0-6AER-6KLQ

[19] w3resource. (n.d.). https://www.w3resource.com/c-programming-exercises/conditional-statement/c-conditional-statement-exercises-8.php ((Accessed on 07/20/2020))