Teaching Intro Programming Using VR: Mental Models for C-language Interpretation

Tajah M. Warren
Rhodes College,
Memphis,
Tennessee
wartm-21@rhodes.edu

September 12, 2019

Abstract

Students that get into computer science have a difficult time understanding and accessing C-language code. The reason for this is because they do not have the correct mental model to properly understand the processes that happen when a program runs. This lack of understanding causes students to feel inadequate and to leave the field of computer science altogether. To find a potential solution to this problem, I will examine the current mental models and look at what the disconnect is between traditional teachings and how to improve the way C-language is taught while using VR to better implement the correct mental model. The focus of the research this time is to find a better way to teach memory allocation and to see if the traditional teaching method is better in comparison to the new method that was developed using VR. With this new method, students should have a better understanding of C-language code and feel more confident in their abilities so they continue to stay in the computer science field.

1 Introduction

In this paper I will explore why students have difficulty understanding C-language code and to look into different ways that can help students become better C programmers. C-language is one of the most difficult programming languages to learn. Many students are not able to grasp the correct mental model that is needed to properly understand what a C program is doing when it runs. Learning C- language without a proper understanding of a correct mental model can demotivate students and cause them to not want to continue exploring computer science. I want to solve this problem so there can continue to be a steady flow of future computer scientists.

From what I have observed and read about, the problem with retaining students in computer science comes from their lack of confidence in their ability to understand the C-language code given to them or being able to write it. This lack of confidence comes from their inability to understand how C-language works. The problem is that students do not have the appropriate mental model stored away in their minds. What I am looking to do is to find an improved way for students to learn this mental model. One of the elements that I am looking into is using Virtual Reality for its' immersive properties. Immersion can help students understand with a more hands-on approach, giving them the confidence they need to continue with computer science. However, before I can develop the new method of teaching the correct mental model, I must first explore the traditional teaching methods and see the results of those teachings in order to know what to improve.

By looking into the traditional methods of teaching the appropriate mental model (lectures, books, tests/exams/ etc.), I can look for things that I can use in my new method (a Virtual Reality program name TIP VR:Teaching Intro Programming using VR) and discover what elements I need to add in order to make it a more efficient way of learning C-language. This research will impact both students and professors by giving them both a new source to learn and teach C-language in a more efficient way; students will be able to learn using immersion and teachers can have an option that helps explain C-language and its' nuances in a different way for students who are having a difficult time understanding.

This problem of students not understanding C-language and therefore, leaving computer science because of lack of confidence, has not been really addressed in this way before. Professors and researchers have looked into reasons why students are leaving, but have not really tried to look into incorporating new methods of teaching C-language. There is not a current effective solution to helping students understand C-language and improve self-efficacy.

That is why this research is important. Finding a way to implement and use the immersive nature of VR to teach students the correct mental model to analyze C-language, and is this case, understand memory allocation, would make students become more efficient at coding and help retain high numbers in the computer science field. This research is different from others because it has not been explored before and could potentially change the way that professors teach their beginner C-language class and how students approach code.

2 Related Work

In paper [1], the researchers approached the issue of students not having the aptitude to program by trying to come up with an early predictor that determines if a student will be able to successfully understand the way programming works. The issue with coming up with a predictor is that multiple professionals have a different ways of determining if students are able to succeed in computer science. The researchers in this paper decided to use the mental model method of predicting if students are going to be successful in the field of computer science.

The methodology the researchers used was to conduct six experiments testing the students to see which mental model each of them used to solve the problems given to them. All the tests that were given were used to see what mental models the students were using and if the choice of mental model was consistent. The ones who used the same mental model were labeled as the consistent group, the ones who used different models for different questions were labeled inconsistent and the ones who failed to answer were labeled the blank group.

The results of the research showed that making a predictor for if a student will be successful in computer science will not work. The results about whether a student has the proper understanding is inconsistent because professionals cannot know the motivations behind why a student is in the field and also if that student has any inexperience in the field. If a student already has the correct mental model then the experiment doesn't work.The research done in this paper does not fix the problem that I am trying to solve because I am trying to work with students who do not have the correct mental model to solve C-language code. I would have to design the VR program to be accessible to first time programmers as well as professionals in the field.

In paper [2], the researchers created an Augmented Reality learning system (will be referred to as AR from this point) to teach students the complexity and abstractness of developing for 3D applications. Students could benefit this AR-based learning system because it allows them to learn the information needed for 3D programming through contextual visualization and learning interactivity. The visualization aspect of this system will allow students to see the use of programming codes regarding their positioning of objects and their relevant relationships. The interactivity aspect of this system is covered by allowing the real-world to become tangible and manipulable for the students. This approach can potentially help students overcome difficulties with 3D application programming by encouraging them to have positive attitudes towards exploration, inquiry, logical thinking, and reasoning.

The methodology the researchers used was an experimental design that was conducted to evaluate the four aspects of AR-based learning system: system usability, learning efficiency, flow experience, and usage perception. Thirty-four intro to programming students participated in this experiment, so they were not familiar with OpenGL. All of the participants experienced the ordinary version of learning 3D applications programming and an AR-Enhanced version of the same thing. The participants were also divided into two groups: Group A and Group B. Group A students used the AR-Enhanced version first and then the ordinary version and Group B used the versions in reverse. This was done to reduce bias and create a more balanced treatment order. The students were given a short period of ten minutes to be taught how to use the systems. They were also asked to complete seventeen exercises based on the material they were just taught. The group were then made to use the systems, in the order that was described above. They used each system for twenty-five minutes each. After using both systems, the students were given three questionnaires to fill out about flow experience, usage perception, learning efficiency, and system usability. This data from the the surveys was then used in data analysis.

The results of this research concluded the AR-based learning system enhanced the system usability (because of the ability to display virtual information in the context of a real environment), increased the ability to complete exercises in full (because of the additional support of command cards and the holistic window that offered a global view to know what the students did), created a better flow experience (in terms of concentration, autonomy, and immersion), and gave students a more enjoyable experience with learning OpenGL material in terms of efficiency, ease-of-use, and accessibility. The implications of these results show that the AR-system contributed to the betterment of the students perception and flow experience when learning 3D applications material.

Even though this solution worked in the context of 3D applications program learning, using AR-based learning will nor necessarily work in the case of teaching students in an intro to programming class how to properly analyze C-language code. I am trying to solve the issue of how to teach students memory allocation in a VR environment that allows students to interact with memory as they learn. The AR-enhanced system does not give the amount of immersion that is needed to solve the problem of teaching students the proper C-language analysis and mental model.

3 Approach

Some of the unaddressed issues with the current approaches to teaching memory allocation in C-language is that 1) professors do not take into account that students learn in different ways and do not take that into consideration and 2) students will not go to professors and ask questions about what they do not understand if they do not comprehend the material enough to begin with. The current approaches fail to instill students with confidence in their abilities to correct access memory allocation. They also fail to thoroughly explain the correct mental model for C-language analysis.

The new approach that is being proposed is creating a game-like program that uses Virtual Reality to teach students C-language. A game-like program that teaches memory allocation can help simulate what the process is in tangible manner. The program will allow the student using it to go through the process of memory allocation by being able to walk through a computer and going through the process of how a stack allocates memory and stores input from code. The VR aspect allows for the user to be immersed in the environment that resembles the inside of a computer and to be able to interact with elements that make up a computer. To create this program, Unity and C-sharp was used in the making of the landscape that the program takes place in and the 3D open-source modeling program Blender was used to create the models that the user interacts with used in the landscape.

The success of this approach has yet to be determined because this program is the first of its kind to be designed and utilized in this area. There are no other approaches that use a VR program to teach students C-code and help improve their confidence in their abilities to code and raise their self-efficacy levels. The expectation for this program is that it teaches students C-code in a more efficient, faster, and affective manner while also keeping students in the computer science field.

4 Experiments

The VR program will take the student using it through a streamlined, concise, but clear lesson about how memory allocation works in C-language. Students will understand memory allocation and have the correct mental model to analyze C-language code correctly and quickly. We first must compare the new VR program and its' results to the results of the traditional teaching methods (reading a lesson on memory allocation in a book, lecture from a professor, etc.) The current consensus is that the traditional teaching methods are not providing students with the accurate mental model needed for analysis of C-language code.

The TIP VR program results will be compared to the traditional way of teaching memory allocation. The method we are using to get the results needed for the comparison is using a practice exam. We hope to find that the new method (TIP VR) produces more understanding in students versus the traditional teach and test way.

Before I can conduct the experiment, these the the assumptions I must make:

- The student has little to no experience in coding.
- The student is interested in computer science as a major choice and career choice.
- The student will be getting no additional outside help with the exam.

These are all obstacles I must keep in mind when designing and executing the experiment. I will be testing if the traditional methods of teaching memory allocation are better than TIP VR by hosting a lecture that teaches memory allocation the traditional way, then giving a sheet a practice problems for the students to complete. The answers that the students give for the problems will be compared to the correct answers and from that comparison I will be able to determine what mental model the students are using to solve the problems and what needs to be the focus in the TIP VR program to help better explain C-language memory allocation.

Physical experiments are necessary in this case because we need to test the traditional methods and compare them to the new one (TIP VR). One hundred students are needed to get enough results to compare to the TIP VR results. I will run the experiment five times, twenty students per experiment. New software and hardware is not necessary for this experiment since it is only using a lecture to teach memory allocation and the problems will be given on sheets of paper. I will be measuring whether the results of the traditional methods are better than the TIP VR results and if the TIP VR program improved students understanding of C-language and their confidence in their abilities to code.

5 Future Analysis

I hope to see that the data from this experiment shows that students have different, incorrect mental models when taught the original way and that the TIP VR program improves their understanding of memory allocation. This improving of their understanding raises their confidence in their abilities, which in turn makes them stay in the computer science field to go on and be researchers, programmers, etc.

The results of the experiment should be different from the results of the VR program. The results of this experiment should show that the traditional methods of teaching are less efficient than the VR program and open to misinterpretation about mental models. Students will have multiple answers to the problems given to them because they do not understand how memory allocation works. Some of the students will get the questions right by either guessing, understanding the material partially, or understanding fully, but the answers will be inconsistent.

Data being surprising cannot be be determined yet because the experiment has not been conducted. When the experiment is carried out, that is when data analysis can be properly done. The data should show that students have different answers because they have different understandings of memory allocation. The outliers in the data will most likely be the correct answers that were made by students who either 1) guessed well or 2) understood memory allocation correctly after being taught the lesson in the traditional way.

There should be no other factors other than the ones stated in section 4. If there are any other factors that could affect the data, they have not been discovered yet. Other factors could be discovered when future experiments are added to test this program.

6 Conclusion

Students do not have a great sense of self-efficacy because they do not understand how to interpret C-language. Students leave the field of computer science because of lack of confidence. The problem is not their ability to understand the material that is being taught, but the way that the material is presented. Many students do not have a learning style that is compatible with the traditional methods of teaching C-language code, and in this case, memory allocation.

To fix this problem, a new program called Teaching Intro Programming in VR (TIP VR) will teach students memory allocation using a more hands-on, tangible approach that shows them a step-by-step guide on how the input they type in is stored onto a stack. In this paper, the experiment that will be conducted is to get data to look at for comparison to after TIP VR is introduced. This experiment will use traditional teaching methods to teach a lesson in memory allocation to students and then give them a practice exam for them to answer problems using the information that they just learned. The results will show mixed answers because many students will not completely understand the concept of memory allocation in C-language.

These results imply that the students all have different mental models when it comes to understanding memory allocation in C-language, which in turn translates to other aspects of the students understanding C-language. This misunderstanding causes students to doubt their abilities in becoming programmers and in turn causes them to leave the computer science field. When they leave the field there is now a smaller amount of potential computer science researchers, which means less progress and innovation towards and in the field. For future extensions, more lessons in C-language should be added to the program. Right now, the only thing available is a lesson on memory allocation. The program should also be tested in a more realistic environment, like an actual classroom, instead of just a controlled environment with just a certain amount of students.

We have to look into why the traditional methods of teaching C-language to students do not work. Some students learn better by being put into an immersive environment which is what I hope is provided with the TIP VR program. I want this approach to the issue to be incorporated into the curriculum for beginner computer science students so that they can have different options when it comes to learning how to code. Having this new method can improve students' issues with not only understanding C-language, but also improve their confidence in their abilities to code in order to continue to have a steady flow of potential computer science researchers and programmers.

References

[1] Richard Bornat and Saeed Dehnadi. "Mental models, Consistency and Programming Aptitude". In: *CRPIT* 78 (2008), pp. 51–61.

[2] Chin-Hung Teng, Jr-Yi Chen, and Zhi-Hong Chen. "Impact of Augmented Reality on Programming Language Learning: Efficiency And Perception". In: *Journal of Educational Computing Research* 56.2 (Apr. 2018), pp. 254–271.