

Journal #1

I had several problems getting used to having to catch my own spelling or syntactic errors which usually occurred because I typed too quickly. My solution was to occasionally compile my project to ensure there were not any significant problems syntactic problems. If I were to wait until the end of writing my code, I could potentially take hours to debug and find errors in larger programs.

Another problem I encountered was simply getting used to the more primitive coding style of C++. There were not as many libraries or resources I am accustomed to accessing to solve some of the more tedious logic or “type” problems. Although after searching through Google, I had found some clever solutions that, in my opinion, solved some of the problems in simpler fashions.

One of my favorite references is a book the University of Utah library gave away. It is “Starting out with C++” 5th ed. One of the reasons I really like the book is because, even though I am an experienced coder, I do not know how to use C++ effectively. This book is designed for novice programmers, but includes many advanced topics. It is the focus on beginners that makes the book easy to read and understand for some of my more trivial questions I sometimes ask.

My first step for preparing for the next assignment is to review the math algorithms needed. What I will try to do is take the given addition and multiplication algorithms and adapt them to what is left: division, subtraction, and modulus. This step will mostly likely be more time consuming than the rest, but I estimate about an hour to have the algorithms ready.

Second, I will plan the basic class structure and any methods I will need. Because the program is small, I assume this will take 30min at the most. The only inconvenience may be to add methods later on that I did not foresee a need for.

Lastly will come the actual coding and testing. Were this not in C++ and in a primitive environment I could do this in less than 30min, but because I am not familiar with C++ or used to needing to catch my own mistakes, this can take anywhere from 30min to a couple hours. The most difficult area will be testing in my opinion.