Joshua Pollock

CS 200 Project 4

3 March 2017

Joshua Pollock

CS 200

Project 4 – Bitwise Ops

3 March 2017

**Purpose:**

The purpose of this project is to create a program that shows the results of various bitwise operation being applied to 2 integers. The results must be shown in a decimal, hexadecimal, and binary value. This project will help to practice our programming in the C and/or C++ language. The program will prompt the user to input two integers (x and y), and print out the results using the bitwise operations: x&y, x|y, x^y, ~x, x>>y, and x<<y. As explained in class we are unable to use the bitset operator, but can browse the internet for solutions. The solutions that are printed should be padded with zeros in order for clean printing.

**Research:**

I didn’t need much research for this project. All that I really needed was taught in class. However, I did conduct some research into bitwise operators, hexadecimals, strncpy, strncat, switch, and case. I saw that the given file, bitwiseBase.cpp, contained these items and wanted to learn more about them in case I needed to implement them into my code. These are the websites I found in my research (the first website was the most helpful for troubleshooting outputs):

<https://www.tutorialspoint.com/cplusplus/cpp_bitwise_operators.htm>

<https://www.mathsisfun.com/hexadecimals.html>

<http://www.cplusplus.com/reference/cstring/strncpy/>

<http://www.cplusplus.com/reference/cstring/strncat/?kw=strncat>

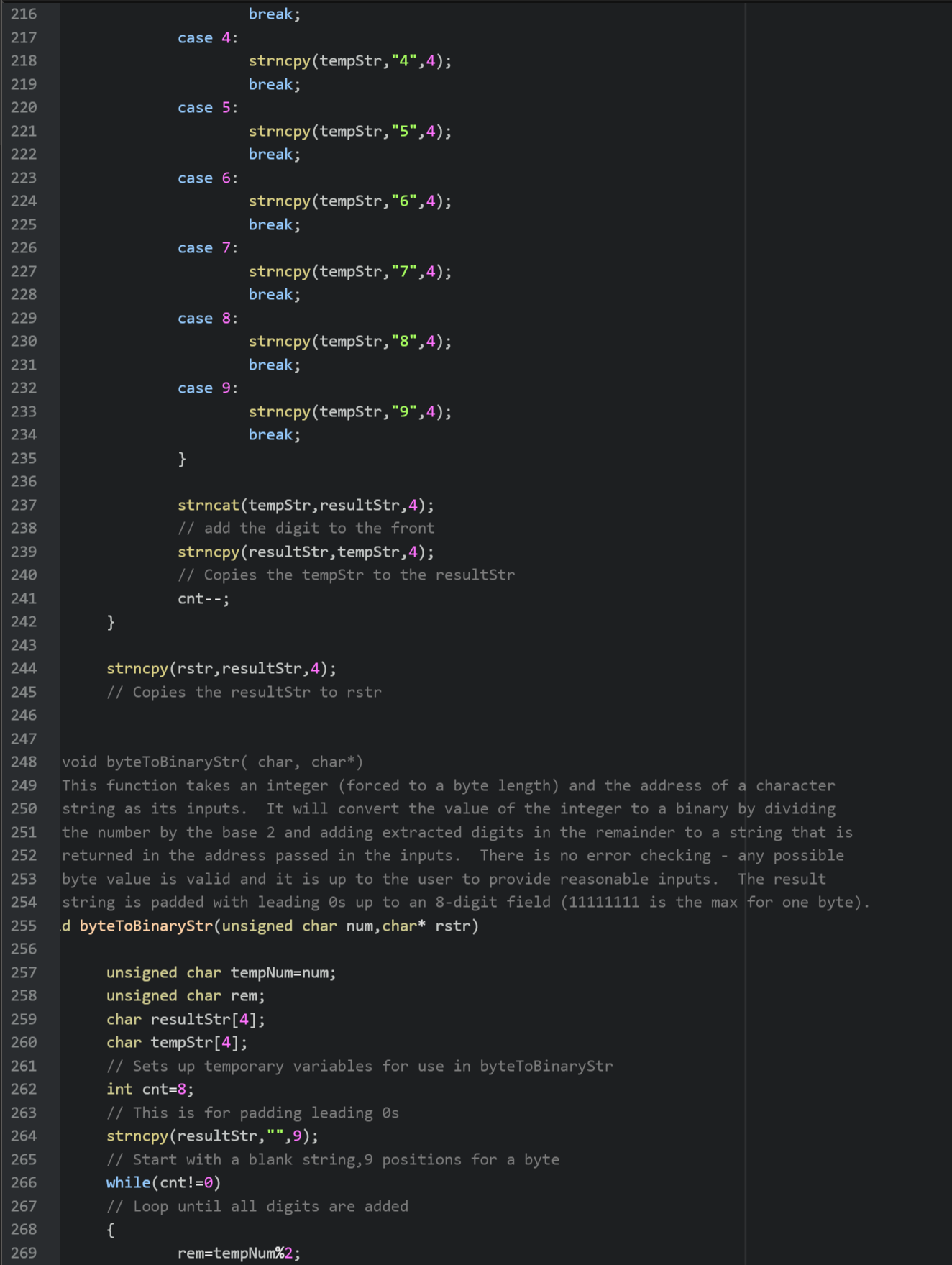
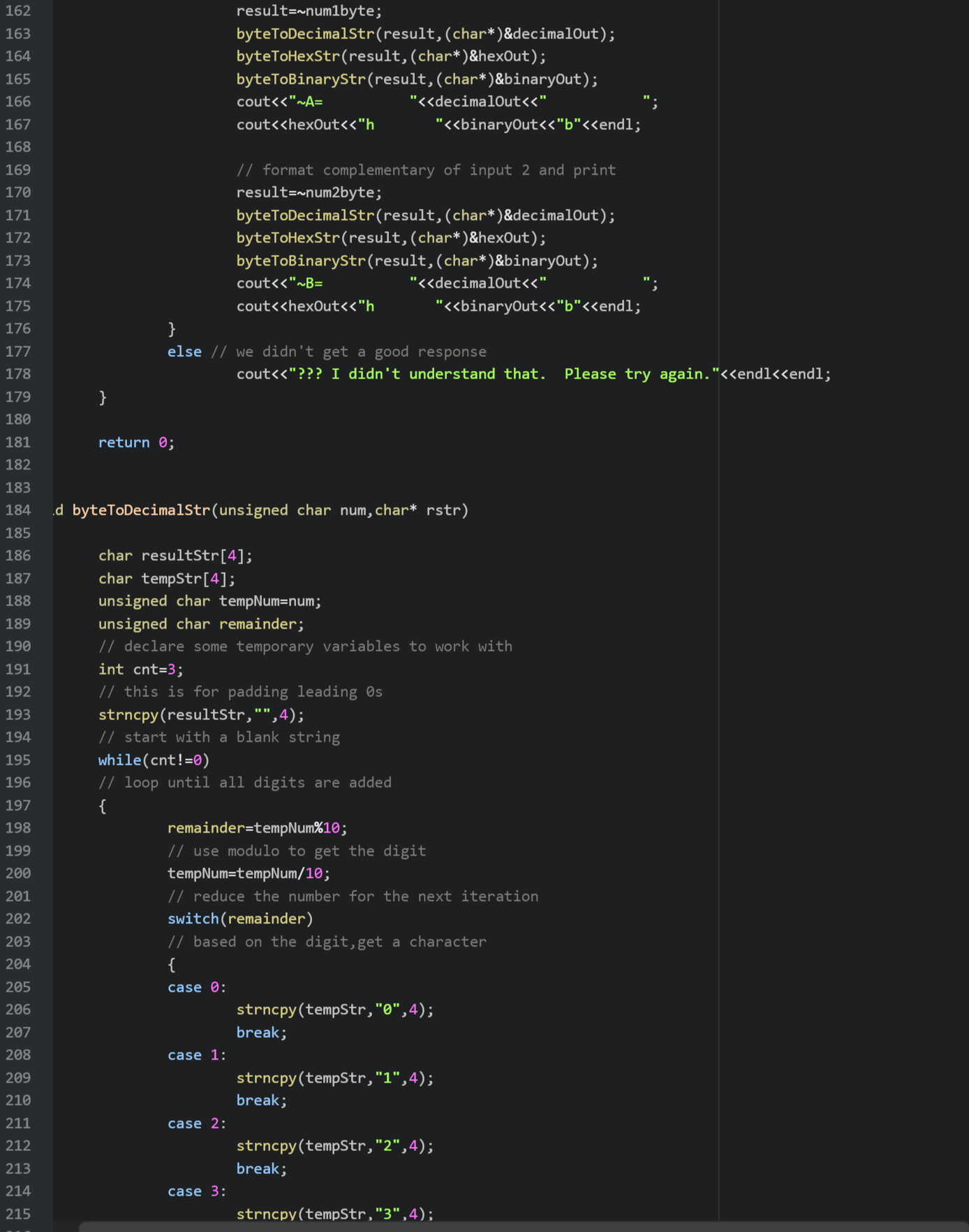
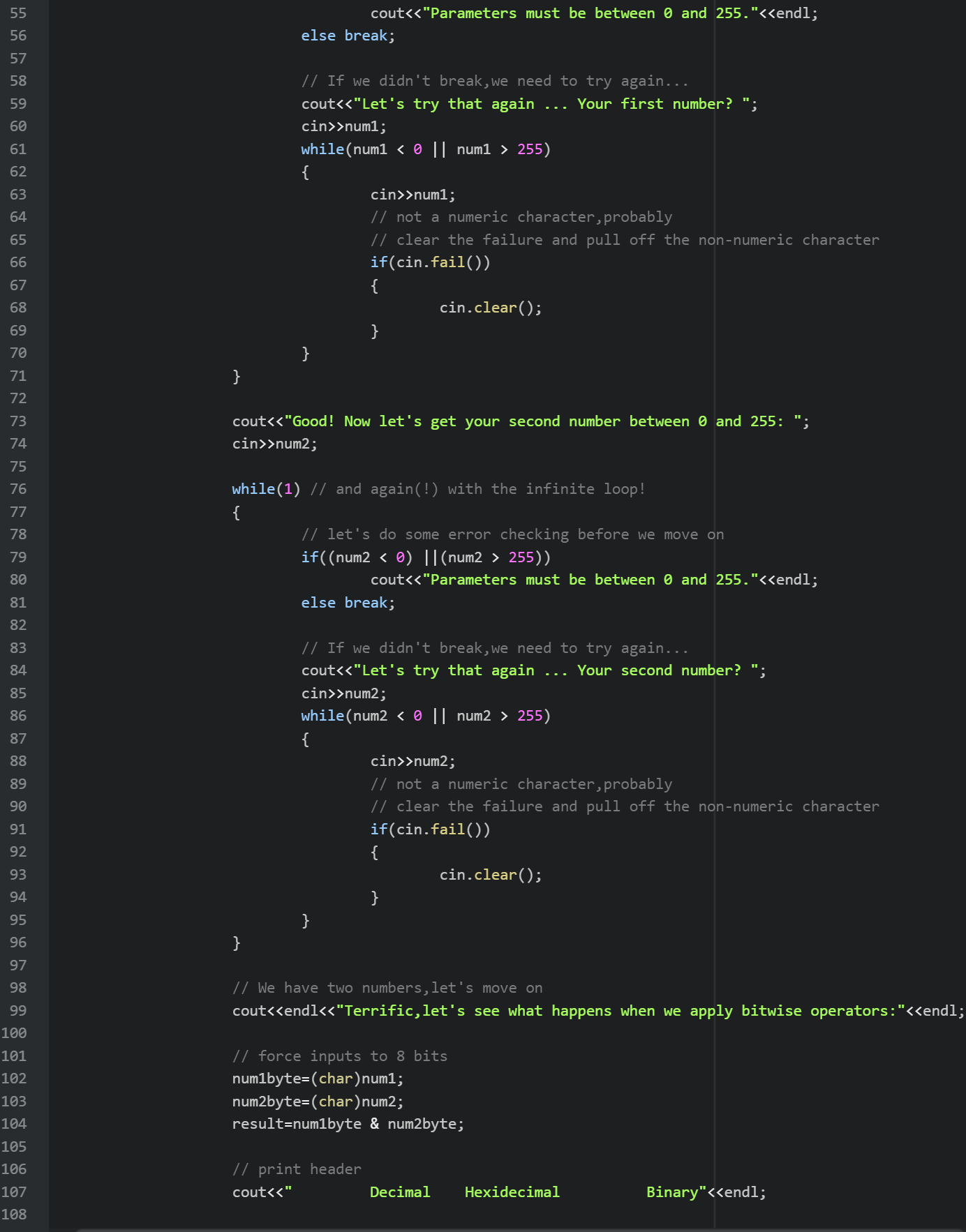
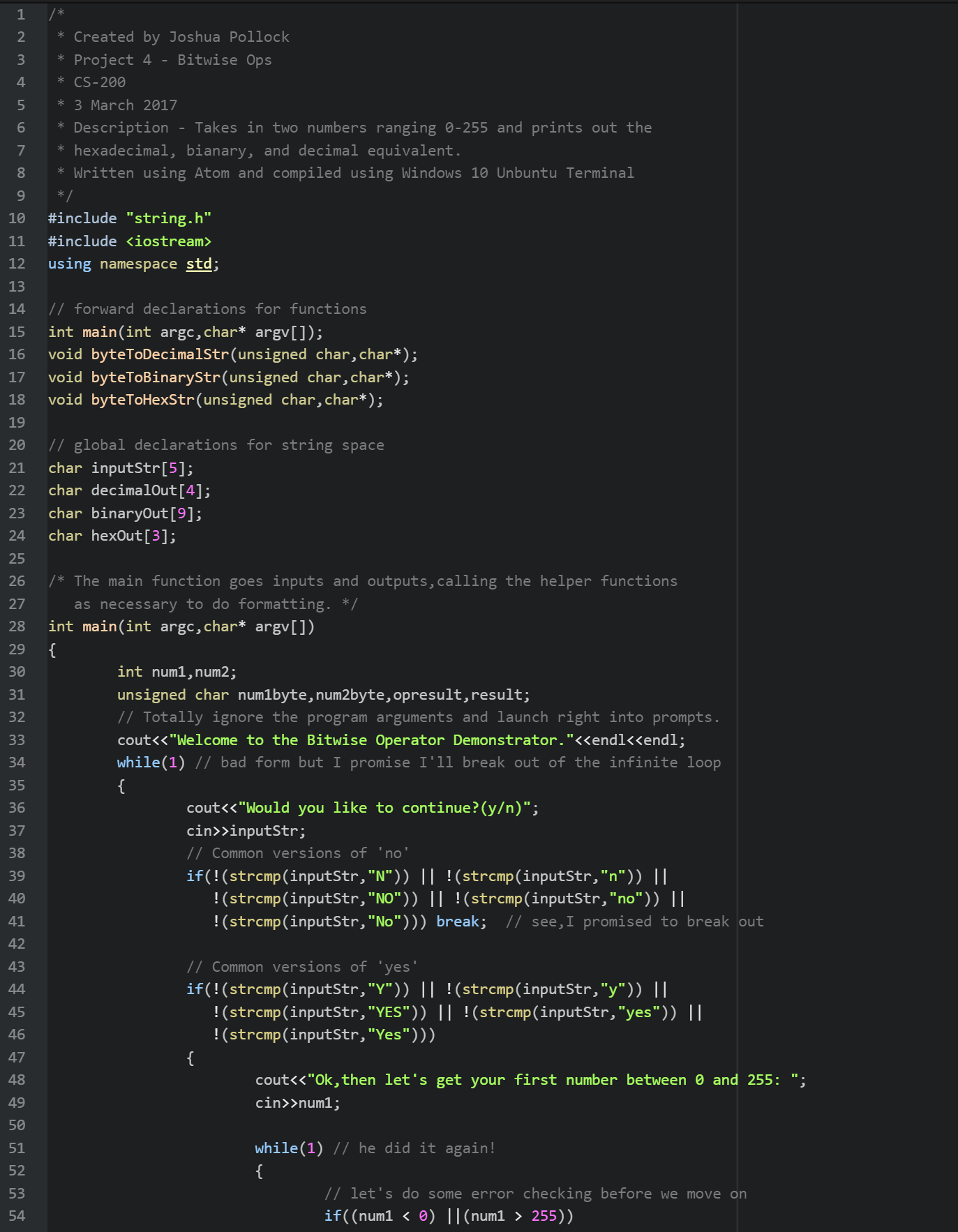
<http://www.cprogramming.com/tutorial/lesson5.html>

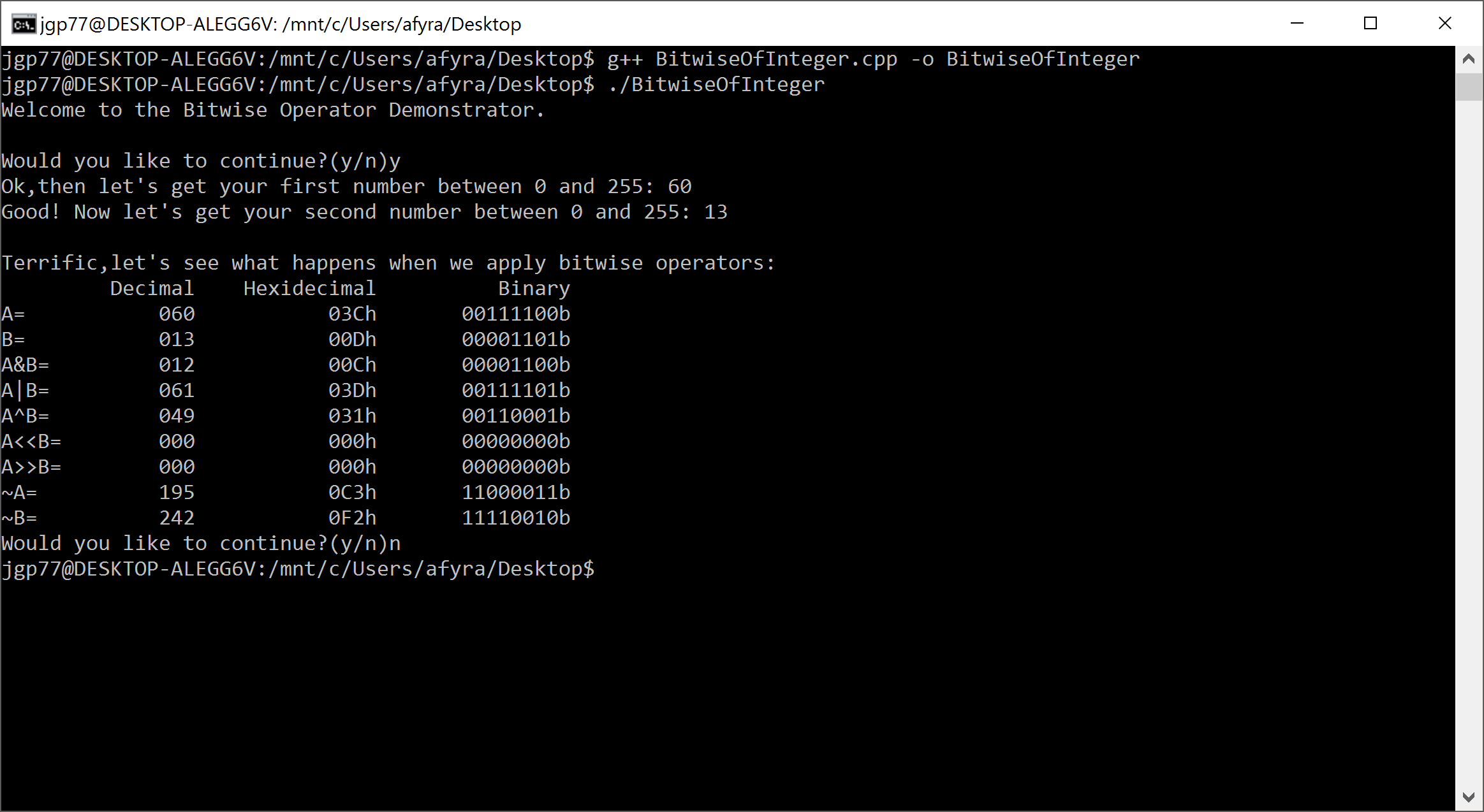
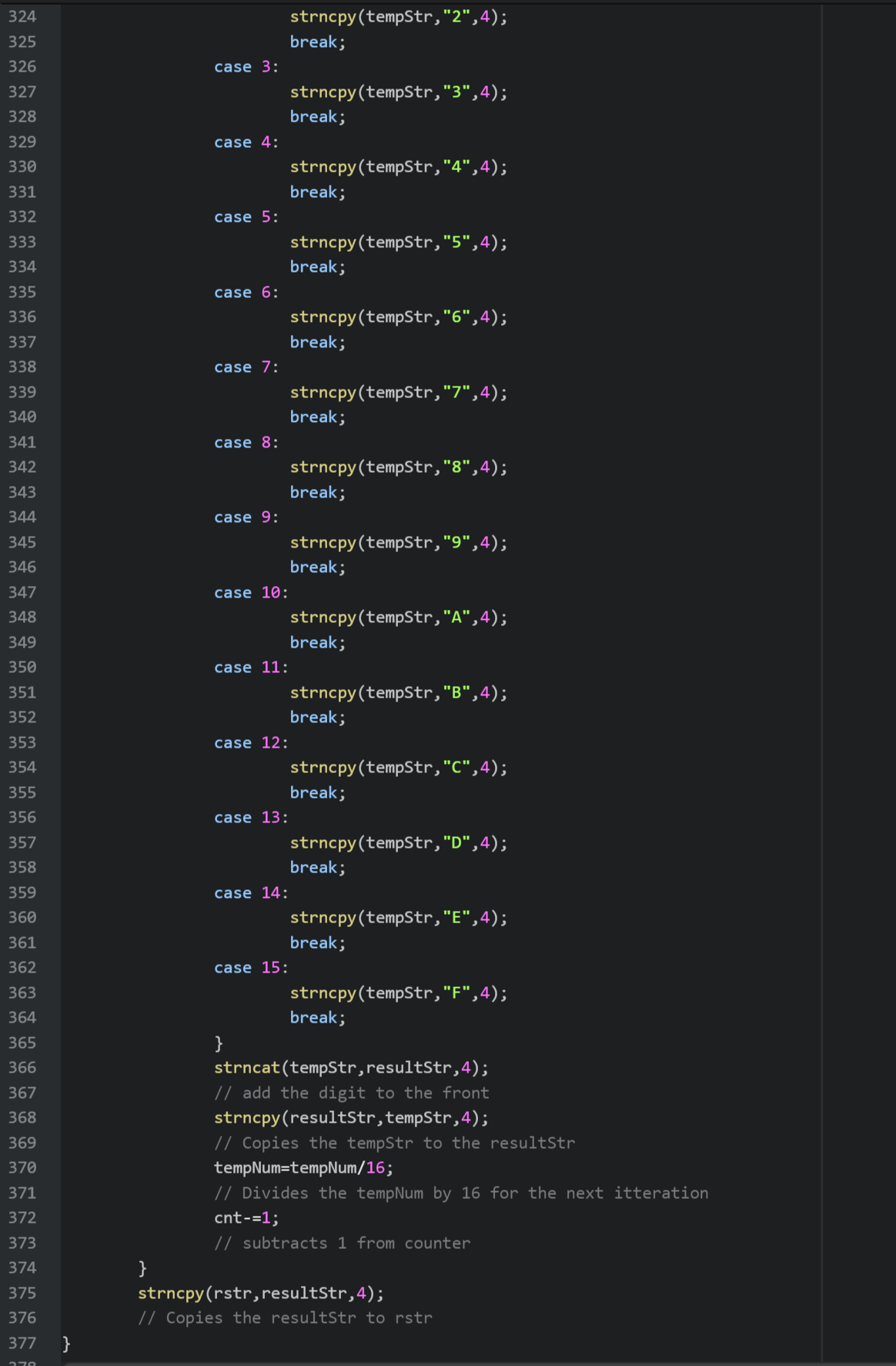
<http://en.cppreference.com/w/cpp/language/switch>

After reading these websites I had a handling on what these functions and operations did, and decided to begin programming.

**Program:**

Below will be screenshots of the the code and also an execution of the code using the values 60 and 13. I found the bitwiseBase.cpp file to be extremely helpful. Looking at the byteToDecimalStr given to us, I was able to base byteToHexStr and byteToBinaryStr off of it. I simply changed around the switch/case statements, the counter, and remainder for the switch. From here all that was left to do is using strncat and strncopy to print out the final result(rstr).





**Conclusion:**

Overall this project was a great experience for me to advance my C++ programming. It turned out to be very simple after classroom explanations and referencing the bitwiseBase.cpp file. The program turned out to be a little bit of copy and pasting from the byteToDecimalStr void. I didn’t mindlessly copy and paste from it however. I did research into what everything meant and it made sense to me. The one thing I am unhappy with was my inability to make the complementary negative decimals instead of positive variant. I attempted many ways to try and get the program to display negative decimals. Due to time constraints and the project constraints, I decided it would be best to move on from this and just have them remain positive numbers. This project helped to expand my knowledge of the C++ language, and it was actually quite fun to see everything coming together.