Mastermind

Victor Alcaraz

48983

10/25/2015

C++

**Table of Contents**

Introduction……………………………………………………………3

Summary………………………………………………………………3

Description…………………………………………………………….3

**Introduction:**

My name is Victor Alcaraz. I am working on this project for my CSC-17A class at Riverside City College. I am a computer science major going on my second semester. I chose the game Mastermind for my first project of the semester because I like a bit of a challenge but something that is not overbearing. Along with just programming the game, I decided that I would like to add some more elements of technology to the program. This project is important to me because I’m hoping to use it to grow as a more efficient and experienced programmer.

**Summary:**

Taking 10 days to program the game itself, along with the added technology, the game proved more of a challenge than initially thought. In the main.cpp, I have accrued 592 lines of code. In those 592 lines of code, I have:

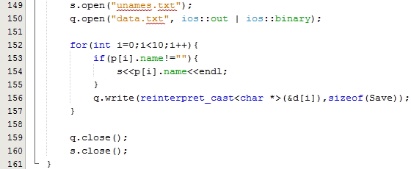
* 31 variables
* 7 functions
* 2 header files (1 structure each)
* 3 allocations of memory (2 arrays of structures, 1 dynamic array)
* 2 objects to write to files, 3 objects to read from file, and
* A whole lot of thinking

To mark things off the checklist of things required in this project, a line number will be shown next to the concept.

* Memory allocation – Lines 47, 48, and 328
  + Deallocation – Lines 78, 79, and 477
* Functions with structures – Lines 92, 110, 142, 172, and 294
* Pointers with arrays and arrays of structures – Lines 47 and 48
* String Objects – Line 372
* Reading and Writing to binary files
  + Reading – Lines 116-129
  + Writing – Lines 150-157

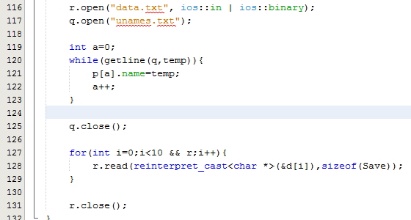
**Description:**

Reading and Writing to binary files took me a while to figure out. I initially tried to send a string along with ints and floats to a single file as binary, but things got a little jumbled and weird as I tried to read the string from the file. So I just resorted to writing and reading ints and floats to two separate files, one binary and one regular. I understood the concept better and was able to save the data from the binary file into the array of structures. Doing this, I was able to save the number of games played, wins, loses, win percentage, and win with least amount of tries per user. That data is in a parallel array with usernames that is written to a file in regular ascii format and read and store it into the parallel array. This will ensure that the correct username is linked with the correct data.



Here are the lines of code that represent writing usernames and data to two separate files. As you can see, everything is within a for-loop to set everything from parallel indexes from two separate arrays of structures. Usernames are saved in the ascii format in the file “unames.txt”, and the data is saved as binary into the file “data.txt”.

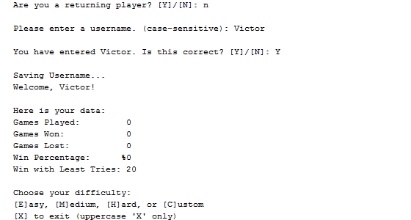
Reading from the file is quite similar except that the stuff coming from the file has to go straight to the arrays of structure with a for-loop and while-loop.



I could get away with using a for-loop for the data simply because the data is read in by size and won’t overlap with the next bit of data.

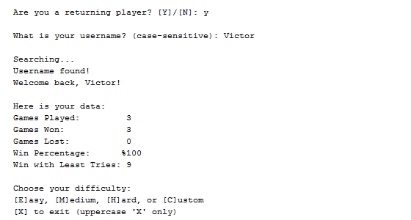
Let’s see how these two methods are in action.

First we will create a new username when prompted.



As you can see, I create a new username “Victor” and confirmed it. Along with that, data is created and set to default values (the win with least tries is set to 20 just to have some comparison when the player gains a win for the first time).

Now after playing the game a few times, I exit the game and rerun it. The first thing the program does is read the files and send the data to the appropriate structure. Once that is done, the prompt asks again, “Are you a returning player?” This time I enter a “Y” and enter the username I used last time. Once the program finds the username, it displays my data from the last time I played and continues to add to it the more I play. Here is a snippet of that.



You as the user can go check out that username as it is still saved in the file. I cannot manipulate the data simply because the file is in binary form. Now is the chance for you to go and create a username and enjoy all your saved data for days to come.