

Assignment 10 & 11 Report

Since the bulk of assignment 11 was experimental/learning, I am combining the 2 reports into one.

Exercises

List of Files:

A10E1.cpp

A10E2.cpp

A10E3.cpp

A10E4.cpp

A10E5.cpp

A10P1.cpp

A11E1-f.o

A11E1-f.h

A11E1-f.cpp

A11E1-g.h

A11E1-g.o

A11E1-g.cpp

A11E1.cpp

A11E2 (The makefile)

A11E3.cpp

A11E4.cpp

Projects

Understanding

I don't think there was anything too in-depth about the understanding of this week's project. But, I will break it down as I understood it.

We (and I apologize for always using the "we" term instead of I. But saying I over and over makes me feel like I sound like a stuckup jerk, so I'll stick with we) want to make a gaming project that combines all of our previous games into one. In previous assignments we made a number guessing game, a word guessing game, and a phrase guessing game.

In this game, we want the user to come on and select which of the 3 games they want to play, and subsequently play them and decide if they want to play again or not. The major change however is that all of the games need to be self-contained within their own classes.

Design

I attached my original drawn workflow to the bottom of this document. Overall the design wasn't too in-depth because most of the code was pre-written out. Each game was previously done and submitted and working, so the only changes that had to be made was to place them into classes instead of just functions.

I made a menu at the beginning outlining the possible gaming choices and choosing one automatically launches the game. Once the user plays the game, they then are prompted if they want to play again.

Changing from a program to a class actually wasn't bad at all. I took all of the existing function code and placed them into the public block, with their variables being in the private block, and just created a new public function in their class to handle what I had previously in the main functions for each program.

Testing

All major testing for each game was done previously with each assignment I had to do, so I didn't do anything too in-depth in regards to the gameplay itself. Most of the testing fell on the selection of the games and the replaying options.

Reflection

Nothing much in regards to Assignment 10 as most of it was review from the previous assignment (I assume we are jumping into the CS 162 side now).

The biggest thing I noticed was how much easier error handling was if I used a string to handle all inputs. For example, in A11E5 I had all of the inputs in regards to which function and which function type as a string. By doing so, I didn't have to worry about cin error handling because the string would accept any input as valid; then I could just check on it and have the user enter a new value. Granted, I doubt this is not the most efficient way in terms of memory usage and processing speed; but never having to worry about cin.clear and cin.ignore is quite nice. I used that same idea in the project for selecting which game the user wants to play.

CS165: Assignment 10

5/19/14

Program

- Asks user what game
- plays game
- asks if they want to play again
- each game contained inside a class

