Homework 4

C++ and file output

February 1, 2020

Abstract

The objective is to give you practice using C++. It poses the same problem as HW3, but you need to solve it in C++.

Be sure to construct test cases for each function used in the production code. Put thought into the test cases. The goal is to know that a function works, when it passes all of its test cases. One test case per function is usually not enough.

- Construct a sequence diagram for your project. You can draw it by hand and include a phone/photo, or draw it with a software tool. Note that, depending upon whether you set values for attributes of your object or not, the arguments/parameters that appear in messages may change from those in your HW3 sequence diagram.
- Use the test-driven development style for developing your code. Document this as described in HW3.
- HW3 required reading data from a file. This homework also requires writing a file.
- In addition to printing the rooms that have been searched, and the treasure subtotals, as they are accumulated, as you did for HW3, please also create an output text file, readable by, for example, Notepad, and write that information into the file.

Things to do:

1. Either:
   (a) Make a C++ project from the Hello,World project.
   (b) Populate that project with tests.cpp, tests.h, production.cpp and production .h.

   or use the starter code.
2. Besides using C++, the requirements are the same as in HW3, plus the creation of the output file.

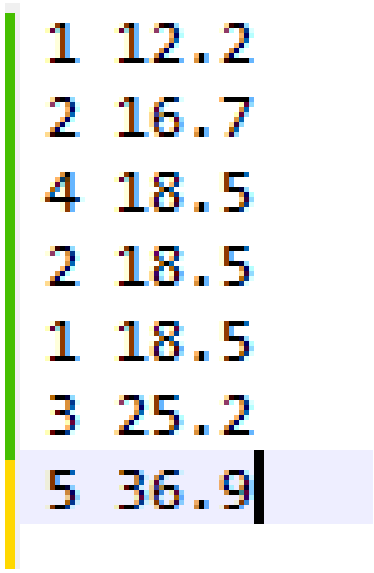

Figure 1: The output file contains multiple lines, each of the form of one integer followed by one double. The integer designates the room the searcher is in, and the double designates the accumulated treasure.

Grading

Criteria	Possible Poin
Project that looks like starter code	20
Sequence diagram that reflects the problem statement	20
Documentation of code development (those screenshots) that clearly follows test-driven style	20
File of rooms searched, visited and treasure collected subtotals, that correspond	20
Correct treatment of input files and command line arguments	20
Total	100