CSCE 240 – Exam Three

Due: 11:59pm on Tuesday, December 6. **Late exam submissions will not be accepted.**

This is an exam. As you work on these problems, you may use your textbook, class notes, and the recorded lectures. You may ask your instructor clarifying questions. You are not to discuss the problems with other students or seek help from other individuals. All work submitted must be your own. All code submitted will be examined for plagiarism and violations will be reported to the office of Student Conduct and Academic Integrity.

Test all of your code on a Linux lab computer. All source files submitted must compile and run on a Linux lab computer of the instructor's choice. Submissions that do not compile on the Linux workstation will receive no compilation or execution/correctness points.

Problem 1

Deliverable: *problem1.h*

Purpose: Write a template function named *FindAndReplace* that takes an array of values, the size of the array, a value to find, and a value to replace as arguments. The function should replace all occurrences of the third argument in the array with the value of the fourth argument.

Specifications:

- The files for this problem are in the attached *problem1.zip*
- Implement your function in *problem1.h*. This is the only file to be submitted to the assignment for this problem.
- A makefile has been include to compile and run the initial tests provided in *testproblem1.cc*

Initial Testing:

Initial tests have been provided in *testproblem1.cc*. You are encouraged to implement more rigorous tests. Your program will be graded with a modified version of *testproblem1.cc*

Points:

style: 1 point

documentation: 1 point

clean compilation: 1 point

execution / correctness: 1 point

Problem 2

Deliverable: *set.cc*

Purpose: Add the *AddElement* member function for the Set class. The prototype and more details regarding the requirements are on lines 51-56 in the attached header file, *set.h*. The attached files for the *Set* template class are similar to the files created during our class meetings.

Specifications:

- The files for this problem are in the attached *problem2.zip*
- No changes should be made to the attached *set.h* header file.
- Add all of your code to the attached *set.cc* file, and attach your updated *set.cc* file to the assignment. This is the only file to be submitted for this problem.
- A makefile has been included to compile and test your code with the initial tests included.
- Ensure that your class will compile, link and run with the initial tests provided in order to earn compilation points and to be eligible for correctness points.

Initial Testing:

testset.cc has been included with some minimal initial tests for your function. You are encouraged to include more rigorous tests before submitting your class. This problem will be graded with a modified version of this file.

Points:

style: 1 point

documentation: 1 point

clean compilation of *set.cc*: 1 point

***AddElement* passes instructor tests:** 3 points

Problem 3

Deliverables: *geographiclocation.h, geographiclocation.cc, historicsite.h,* and *historicsite.cc*

Purpose: Create a *GeographicLocation* base class with private data members for the latitude (a double between -90 and 90) and longitude (a double between -180 and 180). More detailed requirements for the *GeographicLocation* class are included in the comments in the attached *geographiclocation.h* header file.

Also, derive a *HistoricSite* class with the *GeographicLocation* class as the base class using public inheritance. The *HistoricSite* will have a string for the description of the site and an int for the year the historic site was established as private data members. More detailed requirements for the *HistoricSite* class are included in comments in the attached *historicsite.h* header file.

Specifications:

- The files for this problem are in the attached *problem3.zip*
- Implement the classes in the *geographiclocation.h, geographiclocation.cc, historicsite.h,* and *historicsite.cc* files provided, and attach your revised files to the assignment. These are the only four files to be submitted for this problem.
- A makefile has been included to compile, link, and run the included test files with your code. Ensure that your classes compile, link and run with the initial tests provided in order to earn compilation points and to be eligible for correctness points.

Initial Testing:

- A makefile has been included to aid in using *testgeographiclocationconstructor.cc, testsetlatitude.cc, testsetlongitude.cc, testhistoricsiteconstructor.cc, testsetdescription.cc, testsetyearestablished.cc, testsites.cc* and *checkit.cc* to test the basic functionality of your classes. To use the makefile, your directory should include:

  your files: *geographiclocation.h, geographiclocation.cc, historicsite.h,* and *historicsite.cc*

  the makefile: *makefile*

  the test files: *testgeographiclocationconstructor.cc, testsetlatitude.cc, testsetlongitude.cc, testhistoricsiteconstructor.cc, testsetdescription.cc, testsetyearestablished.cc, testsites.cc* and *checkit.cc*

  the subdirectory "output" that contains *correctglprint.txt, correcthsprint.txt,* and *correcttestsites.txt*

- To test your *GeographicLocation* constructor, type:
  make testgeographiclocationconstructor
- To test your *SetLatitude* function, type:
  make testsetlatitude
- To test your *SetLongitude* function, type:
  make testsetlongitude
- To test your *HistoricSite* constructor, type:
  make testhistoricsiteconstructor
- To test your *SetDescription* function, type:
  make testsetdescription
- To test your *SetYearEstablished* function, type:
  make testsetyearestablished
- To test the virtual *Print* function for all classes, type:
  make testsites
- Note: the three print tests above run the executable generated by the included *checkit.cc* source file which compares the output created by your functions to the expected output (held in the files provided in the output folder).

Points:

style *geographiclocation.h*: 0.25 point

style *geographiclocation.cc*: 0.25 point

documentation *geographiclocation.h*: 0.25 point

documentation *geographiclocation.cc*: 0.25 point

clean compilation of *geographiclocation.cc*: 1 point

execution / correctness of *GeographicLocation* constructor: 0.5 point

execution / correctness of *SetLatitude*: 0.5 point

execution / correctness of *SetLongitude*: 0.5 point

execution / correctness of *GeographicLocation::Print()*: 0.5 point

style *historicsite.h*: 0.25 point

style *historicsite.cc*: 0.25 point

documentation *historicsite.h*: 0.25 point

documentation *historicsite.cc*: 0.25 point

clean compilation of *historicsite.cc*: 1 point

execution / correctness of *HistoricSite* constructor: 0.5 point

execution / correctness of *SetDescription*: 0.5 point

execution / correctness of *SetYearEstablished*: 0.5 point

execution / correctness of *HistoricSite::Print()*: 0.5 point

compiles / links with instructor's modified *testsites.cc*: 1 point

execution / correctness with instructor's modified *testsites.cc*: 1 point