**CS 3140 – C++ Programming Language**

**Final Assignment (250 points)**

Course instructor: Patrik Boloz

Deadline: 5/12/2023 by 11PM (uploaded to Brightspace)

Your task for this final assignment is to produce computer code in the C++ language and upload all required/generated files to Brightspace by the deadline. Please read the instructions carefully and follow them to get the full number of points. To achieve full points, each numbered assignment is worth 50 points, therefore completing all totals 250 points. Any code uploaded that does not successfully run will be considered incomplete and will result in only partial point assignment, which will depend solely at my discretion. The use of AI tools (like ChatGPT) and straight up copying of code from the internet will be considered as cheating, giving automatic 0 points for this assignment.

The numbered assignments are as follows:

1) Write a program to use the loop to find the factorial of a user given number (use the cin function) between 0 and 10. If the number is negative or bigger than 10, print out a message saying that the calculation cannot be performed. This factorial should be implemented as a function that returns the factorial number. Print out the result in the terminal.

Hint: Use if else conditions. The factorial symbol: ! means to multiply all whole numbers from the chosen number down to 1. For example, if a user inputs 5, then the following calculation will happen: 5! = 5 \* 4 \* 3 \* 2 \* 1 = 120.

2) Write a program in C++ to convert temperature in Celsius to Kelvin which also saves this information in a text file (use an appropriate file name). This conversion will be created as a function and will return the converted float value. Print out the conversion to the terminal together with the user inputted value and also save this output information in a text file.

Hint: To get Kelvin from Celsius, use the following formula: Kelvin = Celsius + 273.15. To use file input and output, use the fstream library.

3) Without the use of “using namespace std;” create a function called void weather\_report(string name) that will take one parameter called name and will greet that name. Also create a string array of different weather situations and print out the current weather.

Hint: Use the random and ctime libraries to generate a random index which you will use to get a random value in your string array. Also, if you have trouble with not using the “using namespace std;”, try it first with the line, run your code successfully, then remove it, and fix the code.

4) Create a Hotel class, that will have one parameter in the init method called name, so whenever the hotel object is created, you specify the name of the hotel. This hotel will also start with int number of guests 0. Create three methods:

1) add\_guest – this method will increase the number of guests by 1 and will print out a message that the specific guest has checked in.

2) check\_out\_guest – this method will decrease the number of guests by 1 and will print out a message that the specific guest has checked out.

3) get\_numberOfGuests– this method will return the number of guests.

Then test out your new class by creating an object, add 4 guests, print out the current number of guests, check out 1 guest, and print out the final current guest number.

5) Create a parent class called Building and 2 child classes that inherit from the parent class Building called Storage and Corner\_Store. The class will have these attributes: string name, string size, int opened, int closed. The Building parent class will have a constructor that will take a string building\_name (name of the building). This constructor will set the size to “Medium”, opened to 9, and closed to 5. You will create these following methods:

1) set\_open\_hours(int hour) - this method will set the attribute opened to the hour the store will open. It will then print out a message saying that the store’s opening hours have been changed and show the hour.

2) set\_closed\_hours(int hour) - this method will set the attribute closed to the hour the store will close. It will then print out a message saying that the store’s closing hours have been changed and show the hour.

3) set\_name(string new\_name) - this method will update the building’s name to a new one and will print out a message saying that the store’s name has been updated and what the new name is.

4) show\_info() - this method will print out the details of the object created from the class: name, opened, closed, and size, in a nice and neat matter.

5) get\_name(), get\_size(), get\_open\_hours(), get\_closed\_hours() - these methods will return their respective attributes of name, size, opened hours, and closed hours.

Then by using inheritance create 2 child classes Storage and Corner\_Store, where the only differences are that Storage will have a default building size of “Big”, opened at 1, closed at 2; and Corner\_Store size “Small”, opened at 8, and closed at 3. Create one object per each class, use at least one setter method, and then use the show\_info() method on all objects to print out the details of each building.