NAME: ____________________________

BU ID: ____________________________

Midterm Exam – May 10, 2022, Tuesday

EC327 Introduction to Software Engineering

Total: 100 points

****MAKE SURE TO READ THE FOLLOWING BEFORE STARTING YOUR EXAM****

Exam Rules:

1. **Save your work often during the exam**. You would not want to lose any work.

2. You **CAN** use the following during the exam:
    1. Your own 1-page cheat-sheet (can be double-sided).
       No sharing!
    2. Linux programs (gcc, vi, gedit, emacs, and so on).
    3. Course material (lecture notes, solutions, etc.) posted at: learn.bu.edu -- EC327 Spring2022
    4. www.cplusplus.com

3. You **CANNOT** use any personal electronic devices (including hardware calculators, cell phones, laptops, tablets, or others) or any other books or printouts.

4. You have until **5:00pm (SHARP)** to finish and submit your exam. Please follow the submission instructions below. The submission site will automatically close after the deadline, so be sure to submit on time.

5. Do not spend too much time on any one question, you can always get back to it later.

6. You **MUST** use GCC 5.2.x and compile with **–std=c++17** otherwise your code may not get any points! To load the correct compiler configuration, please run "**module load gcc**" on the lab machine that you are using.

7. **GOOD LUCK!**

Q1. – Written Questions (20 points)

a. Choose the correct answer among the following ones. **[10 points, 2 per correct answer]**

An external function that has access to private members of a class needs to be declared as:

1. A static function
2. A friend function of that class
3. Privately inheriting from that class
4. An overloaded function

The big-O time complexity of the merge sort algorithm is:

1. O(N)
2. O(NlogN)
3. O(N^2)
4. O(logN)

A function Y with the same name but different prototype than function X is:

1. An overridden function
2. A friend function
3. An overloaded function
4. A redefined function

If I want to have calls to a method invoke a different function depending on the type of object that invokes the method, I should use

1. overloading
2. overriding
3. polymorphism
4. dynamic memory allocation

When inheriting from a parent class in protected mode, the public members of the parent class become ___ in the derived class:

1. public
2. protected
3. private
4. Inaccessible

b. Specify if the following statements are true or false. **[10 points, 2 per correct answer]**

Declaring a method as pure virtual makes a class abstract.	
In Java, you must manually free dynamically allocated memory.	
Private members of a class are not accessible from classes that inherit from it.	
Public members of a class are directly accessible from the main function.	
The binary search algorithm requires that the list it is applied on is sorted.	

Q2. - Files to submit: Q2.cpp. You are given a class Account declared in Q2.h and implemented in Q2.cpp. You are also provided a sample main Q2main.cpp to test your implementation. The file Q2output.txt contains the expected output of your program when implemented correctly. Submit Q2.h and Q2.cpp **(30 points)**

You are given an Account class to represent a bank account. The class declares the following private members:

- unsigned int id, which contains the account ID
- string owner, which contains the name of the account owner
- int balance, which contains the dollar amount available in the account

a) Implement a public method changeBalance(int amount) that adds amount to the account balance (if positive) or removes amount from the balance (if negative)

[6 points]

b) Make changeBalance throw an exception of type underflow_error if the function would bring the balance to a negative value. The exception's message should be "Error! Not enough funds"

[6 points]

Write a class Transfer to represent a bank account transfer. Include the declaration of Transfer in Q2.h and the implementation of its methods in Q2.cpp.

c) Define the following private members for Transfer:

- A pointer to Account source, containing a pointer to the account that the transfer is originating from
- A pointer to Account destination, containing a pointer to the account that the transfer is originating from
- An integer amount, specifying the amount of the transfer. You should not consider the case in which amount is negative

[6 points]

d) Write a constructor with arguments Transfer(Account *source, Account *destination, int sum) that sets the three members of the class.

[6 points]

e) Write a public method void execute() which adds the amount to the destination account and removes it from the source account. execute should catch the underflow_error exception thrown by the changeBalance method in the Account class and print "Error! Not enough funds, transaction aborted". Note that if the exception is raised, execute() should not add the funds to the destination account.

[6 points]

Q3. - Files to submit: Q3.cpp and Q3.h. You are given a base class Employee defined in Q3.h and implemented in Q3.cpp. You are also provided a sample main Q3main.cpp to test your implementation. The file Q3output.txt contains the expected output of your program when implemented correctly. For this question, submit Q3.h and Q3.cpp **(30 points)**

Define a class `Manager` that inherits publicly from `Employee` in `Q3.h`, and implement it in `Q3.cpp`. Manager could potentially have a fixed salary as opposed to an hourly `payRate`.

a) Manager has a private boolean data field `isSalaried`, which is true by default, and a private float data field `salary` that is set to 0 unless `isSalaried` is true. In this case, salary should be set to a positive float. Implement getters and setters for `isSalaried` and salary (see `Q3main.cpp` for details).

[6 points]

b) Implement a constructor: `Manager(string theName, float pay, bool isSalaried).` The constructor should take the value specified by pay and set it as `salary` if `isSalaried` is true, or as `payrate` if `isSalaried` is false. You can set the other member (`salary` or `payrate`) to -1.

[6 points]

c) Override the `pay(float hoursWorked)` function in the Manager class so that it returns `salary` if the `isSalaried` variable is true, otherwise it calls the pay function of its parent class (`Employee`) with the `hoursWorked` argument. This function should ignore `hoursWorked` passed for a salaried Manager.

[6 points]

d) Overload the output stream operator << so that when it is called on an instance of `Manager` it prints out the name, whether the manager is salaried, and either the salary or the `payRate`, so that **`cout << m << endl;`** would print:

`"John Smith is salaried and has a salary of 30000 USD"` in the case `isSalaried` is set to true (30000 is the value of salary), while it will print

`"John Smith is not salaried and has an hourly rate of 20 USD"` in the case `isSalaried` is set to false (20 is the value of `payRate`).

[12 points]

Q4. - Files to submit: Q4.cpp. You are provided a skeleton implementation of your submission in Q4.cpp. Submit Q4.cpp **(20 points)**

This question requires you to use data structures that are implemented by the Standard Template Library.

Write a program to store the book catalog of a library. A skeleton for the program is provided in `Q4.cpp`. The book database is implemented as a map, where the key is the ISBN number for the book (as an integer) and the value is the title of the book (as a string). The user is asked to provide the details for a new book (including both the ISBN and the title) until they press "-1" when asked if they want to insert a new item. A sample run of the program is provided in file `Q4sample.txt.`
Implement the following functionalities:

a) Create a map (with integers as key and strings as values) to store the items provided by the user

[10 points]

b) Every time a new item is provided, first check that a book with the same ISBN does not already exist in the map. If it already exists, print out the following message:

```
A book with that ISBN is already in the library!
```

If the item does not exist, ask the user for the title of the book, and add it to the list. You can assume that the title is composed of a single word.

[5 points]

c) After the user selects that they don't want to add any more items, write the book titles to a file named `library_list.txt`, one item per line. Include only the titles and not the ISBN codes. The program overwrites an existing file if it already exists.

[5 points]

You are also provided a file called `Q4input.txt,` a sample input txt file which you can pipe into your program with "./Q4 < Q4input.txt" and check for the output file (`library_list.txt`) to have the following output:

```
Circe
Dracula
Dune
1984
```

End-of-Exam Checklist

- Turn in your paper exam copy. **Only Q1 will be graded on the paper.**

- Your final directory should be named **<yourBUusername>_<last3digitsofBUID>_final** and contain:
    - Q2.h
    - Q2.cpp
    - Q3.h
    - Q3.cpp
    - Q4.cpp

- **Zip your exam folder and submit** to blackboard (Content – Final – Final Submission).

- **Make sure all your code compiles!**

- **Before logging off, check on the submission portal that your submission is correct and includes all your files!**