

NAME: _____

BU ID: _____

Midterm Exam – March 2, 2022, Wednesday
EC327 Introduction to Software Engineering

Total: 100 points + extra credit (5 points)

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

Exam Rules:

- **Save your work often during the exam.** You would not want to lose any work.
- You **CAN** use the following during the exam:
 - o Your own 1-page cheat-sheet (can be double-sided).
 - No sharing!
 - o Linux programs (gcc, vi, gedit, emacs, and so on).
 - o Course material (lecture notes, solutions, etc.) posted at: learn.bu.edu -- EC327 Spring2022
 - o www.cplusplus.com
- You **CANNOT** use any personal electronic devices (including hardware calculators, cell phones, laptops, tablets, or others) or any other books or printouts.
- You have until **4:30pm (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.
- Do not spend too much time on any one question, you can always return to it later.
- **Question 0** is mandatory and should be done first.
- 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.

GOOD LUCK!

Q0. Submission procedure and finding your way around Linux. [5 points]

****This Question is Mandatory****

- a. Open a terminal window.
- b. Create a folder named **<yourBUusername>_<last3digitsofBUID>_midterm**. For example, student *James Bond* with BU username *jbond* and BU ID *12340007* should type:

```
mkdir jbond_007_midterm
```

**** Make sure you put in your own name and number here, and not those of James Bond!!!**

- c. Change directory to your midterm directory:

```
cd jbond_007_midterm
```

- d. Create test.cpp file in your directory, and write the following code:

```
#include <iostream>
using namespace std;

int main() {
    cout<< "TEST EC327 MIDTERM" <<endl;
    return 0;
}
```

- e. Load GCC with the following command in terminal:

```
module load gcc
```

- f. Make sure your code compiles with the following command below:

```
g++ -std=c++17 test.cpp -o test
```

**** Make sure you run your executable to ensure your code works!**

- g. Zip your directory and submit via the following steps:

```
cd ..
```

```
ls
```

**** You should see your *jbond_007_midterm* directory listed now.**

```
zip -r jbond_007_midterm.zip ./jbond_007_midterm/*
```

**** Make sure to replace James Bond's folder name with your folder name.**

- h. Check that the zip file exists and it has the right contents:

```
ls
```

**** You should see *jbond_007_midterm.zip***

```
less jbond_007_midterm.zip
```

**** You should see your *cpp* files, etc.**

- i. **Submit:**

Go to EC327 Spring2022 Blackboard site → Content → Exams → Midterm. You will see a "Midterm Submission" item. Click on it. You should select your exam zip file

(jbond_007_midterm.zip) through the browser, and then hit submit at the bottom of the page. Check what you submitted. You can always download and check your latest submission.

- j. Now you've learned the submission procedure for this exam. To continue in the Linux terminal:
cd jbond_007_midterm
- ***Or simply, go back to step (c).*

READ BEFORE YOU CONTINUE:

- You are allowed to submit your exam zip file multiple times. **Only your latest submission will be graded.**
- Make sure to follow the folder naming convention outlined above (e.g., *jbond_007_midterm.zip*).
- Make sure to put all the code and all the files you need to submit in your exam directory, e.g., in *jbond_007_midterm*. Download and check what you submitted to make sure your files are there.
- You can find the header files and stub main files needed to complete the programming assignments as a zip file on Blackboard, under Exams > Midterm > Midterm Files.
- Note that you can also access your midterm exam directory on the Linux machine by typing its path in a browser. You can then drag-and-drop files into it.
- **You are not allowed to access any other directories on the engineering grid (such as /ad/eng folders) or ssh to any machines.**
- **Similarly, you must not visit any other course website on Blackboard and strictly use it to view EC327 content.**
- All actions on the lab machines are being logged during the exam. Students not following exam rules may be asked to leave and not receive a grade for the exam.
- **Save your work often!**

Q1. True / False and Multiple Choice [15 points, 5 each]

Write your answers on the exam paper.

1.a.

Determine whether the following statement is **true or false**. If you select false, briefly state **why**.

Given the code snippet:

```
int x = 5;
float y = (float) x;
y += 0.33;
```

The code will return an error upon compilation because y is storing an integer and decimals can't be added to it.

1.b.

Consider the following code:

```
int f = 53;

void foo(int &x, int &y) {
    int *array = new int[5];
    static int st=1;
    for (int i=0; i<x; i++) {
        st += y;
    }
}
```

Which of the following would be stored on the stack? Choose one or more of the multiple choices.

- i. value of f
- ii. address of y
- iii. value of *array
- iv. address of array
- v. st

1.c.

Determine whether the following statement is **true or false**. If you say false, briefly state **why**.

The following C++ program

```
int main() {  
    int i = 0;  
    while(i < 10) {  
        if((i%3)==0) {  
            cout << i << endl;  
        }  
        i++;  
    }  
}
```

Will print the following output in the terminal:

```
0  
3  
6  
9
```

Q2. Short Answers [15 points, 5 each]

Write your answers on the paper exam.

2.a. Briefly explain the difference between pass by value and pass by reference when passing arguments to a C++ function.

2.b.

Convert the following decimal negative number into its 2s complement representation. Show your work when providing the solution (i.e., the steps you followed to get to the results).

-63 (convert to 2s complement, assume a register of 8 bit of size)

2.c.

Convert the following hexadecimal number into its binary representation. Show your work when providing the solution (i.e., the steps you followed to get to the results).

0x2A09 (convert to binary)

Q3. Functions, Arrays, Loops [10 points]

Files to submit: Q3.cpp

Write C++ code implementing the following function.

Submit only the cpp file for the function. Do NOT submit the main function or the header file. You can find the header file on Blackboard (Q3.h), along with a sample main (Q3main.cpp).

**Your code (when compiled with appropriate test code and the necessary standard libraries) needs to compile in order to receive any partial credit.*

Write a function `squareArray` that given an array of integers, prints on the screen the square value of each element, one per line, as follows:

```
void squareArray(int *numbers, int numsize);  
// numbers is a pointer to the input array and numsize is its size
```

For example, if the following array was provided to the function:

```
int array[5] = {0,3,6,5,8};
```

Then `squareArray(array, 5)` would print this in the terminal:

```
0  
9  
36  
25  
64
```

Q4. Arrays and strings [20 points]

Files to submit: Q4.cpp

Write C++ code implementing the following function.

Submit only the cpp file for the function. Do NOT submit the main function or the header file. You can find the header file on Blackboard (Q4.h), along with a sample main (Q4main.cpp).

**Your code (when compiled with appropriate test code and the necessary standard libraries) needs to compile in order to receive any partial credit.*

Write a `removeDuplicates` function that, given an integer array and its size as arguments, returns a pointer to a new integer array containing the values of the original array, after any duplicated values have been removed. The prototype of `removeDuplicates` is as follows:

```
int* removeDuplicates(int* array, int size);
```

For example, after the following code snippet executes:

```
int array[5] = {1,2,3,1,2};  
int *result = removeDuplicates(array,5);
```

The result pointer will point to a new array on the heap of size 3, containing {1,2,3}.

Q5. File I/O [20 points]

Files to submit: Q5.cpp

Write C++ code implementing the following function.

Submit only the cpp file for the function. Do NOT submit the main function or the header one. You can find the header file on Blackboard (Q5.h), along with a sample main (Q5main.cpp).

**Your code (when compiled with appropriate test code and the necessary standard libraries) needs to compile to receive any partial credit.*

You are given a file `restaurants.txt` with the following format:

```
TacoBell  
RaisingCanes  
Otto  
NudPob
```

Each line of the file contains the name of a restaurant (without spaces).

Write a function `inputRatings` that given a restaurant name asks the user to input a rating for that restaurant (a number between 1 and 10), and stores that rating in a file `ratings.txt`. The prototype of `inputRatings` is as follows:

```
int inputRatings(string restaurantname);
```

For example, calling the function as `inputRatings(RaisingCanes)` would produce the following output (items in `<>` indicate the user provided input):

```
Please insert a rating for the restaurant RaisingCanes: <5>
```

Note that each execution of the function should append to the file `ratings.txt` file. In other word if the `ratings.txt` file already exists, the function should add the new rating at the end of it. The format of the ratings file should be as follows:

```
TacoBell 3  
RaisingCanes 4  
Otto 5  
RaisingCanes 5
```

The function should return 0 if the restaurant was found and -1 otherwise. The file `ratings.txt` should **not** be created if the restaurant is not found. The function should return -2 if the user inputs an invalid rating (e.g., something that is not a number or an integer outside of the 1-10 range). You can print error messages if you wish, but these won't be considered by the autograder. For this question you are not required to clear the input buffer if it gets corrupted.

Q6. Mathematical Functions [15 points]

Files to submit: Q6.cpp

Write C++ code implementing the following function.

Submit only the cpp file for the function. Do NOT submit the main function or the header file. You can find the header file on Blackboard (Q6.h), along with a sample main (Q6main.cpp).

**Your code (when compiled with appropriate test code and the necessary standard libraries) needs to compile to receive any partial credit.*

The Riemann zeta harmonic series is defined as

$$\zeta(x) = \sum_{n=1}^{\infty} \frac{1}{n^x} = \frac{1}{1^x} + \frac{1}{2^x} + \frac{1}{3^x} + \dots,$$

Write a function `riemann` that given two unsigned integers `x` and `n` returns the value of the Riemann zeta harmonic series for `x` up to the `n`th element, as a float.

The prototype of the `riemann` function is as follows:

```
float riemann(unsigned int x, unsigned int n);
```

To calculate the power of a number you can use the `pow` function from the `cmath` library. Cplusplus.com is your friend!

Extra credit [5 points]

Files to submit: extra.cpp

Write C++ code implementing the following functions

Submit only the cpp file for the function. Do NOT submit the main function or the header file. You can find the header file on Blackboard (extra.h), along with a sample main (extramain.cpp).

**Your code (when compiled with appropriate test code and the necessary standard libraries) needs to compile in order to receive any partial credit.*

Write a *recursive* function `countDigits` that takes an unsigned integer and returns the number of its digits. The prototype of `countDigits` is at follows:

```
int countDigits(unsigned int num);
```

For example, `countDigits(10329)` would return 5.

End-of-Exam Checklist

- Turn in your paper exam copy. **Only Q1 and Q2 will be graded on the paper.**
- Your midterm folder on the lab machine should contain:
 - Q3.cpp
 - Q4.cpp
 - Q5.cpp
 - Q6.cpp
 - extra.cpp (optional)
- **Zip your exam folder and submit** following the instructions provided.
- **Make sure all your code compiles!**
- **Before logging off, check on the submission portal that your submission is correct and includes all your files!**