

**CST 370 – Summer A 2020**  
**Homework 1**  
**Due: 04/29/2020 (Wednesday) (11:55 PM)**

Name: \_\_\_\_\_ Andrew Bell \_\_\_\_\_

Class ID: \_\_\_\_\_ 1138 \_\_\_\_\_

**How to turn in?**

Write your answer to the questions 1, 2, and 3 and submit it on the iLearn. We **accept only a PDF** file. Do not submit a different file format. In addition, don't forget to **write your name and class ID** at the top of your homework document.

For the questions 4 and 5, you should submit your C++ or Java source files on the iLearn.

So, you have to submit three files (one PDF file and two source files) on the iLearn.

- Note that the due time is 11:55(PM). This is the iLearn's timestamp, not your submission time. Since there could be a long delay between your computer and iLearn, you should **submit early**.

1. Consider the following algorithm to sort an array:

```
Algorithm ComparisonCountingSort( $A[0..n-1]$ ,  $S[0..n-1]$ )  
//Sorts an array by comparison counting  
//Input: Array  $A[0..n-1]$  of orderable values  
//Output: Array  $S[0..n-1]$  of  $A$ 's elements sorted in nondecreasing order  
for  $i \leftarrow 0$  to  $n-1$  do  
     $Count[i] \leftarrow 0$   
for  $i \leftarrow 0$  to  $n-2$  do  
    for  $j \leftarrow i+1$  to  $n-1$  do  
        if  $A[i] < A[j]$   
             $Count[j] \leftarrow Count[j] + 1$   
        else  $Count[i] \leftarrow Count[i] + 1$   
for  $i \leftarrow 0$  to  $n-1$  do  
     $S[Count[i]] \leftarrow A[i]$ 
```

- (a) Apply the algorithm to sort the list 5, 3. In other words, assume that the array **A[ ]** has the initial values 5 and 3. Present the contents of the arrays **A[ ]**, **S[ ]**, and **Count [ ]** after finishing the execution of the algorithm.

A[5,3]

S[3,5]

Count[1,0]

- (b) Similarly, present the contents of the three arrays if the array **A[ ]** has the initial values 9, 3, 3, 9, 3, 6?

A[9,3,3,9,3,6]

S[3,3,3,6,9,9]

count[5,2,1,4,0,3]

2. (a) Find **gcd(140, 310)** by applying Euclid's algorithm. Present the **intermediate steps** of the algorithm clearly.

gcd(140,310)  
gcd(310,140)  
gcd(140,30)  
gcd(30,20)  
gcd(20,10)  
gcd(10,0)  
gcd = 10

(b) Find **gcd(140, 310)** by applying the middle school approach (= prime factors) we covered in the class. Present the **intermediate steps** of the approach clearly.

140 – 70 2 – 45 2 2 – 7 5 2 2  
310 – 155 2 – 31 5 2  
5 2 – 10

3. What is the maximum number of modulus operation(s) made by the Euclid's algorithm for the two numbers  $m$  and  $n$  in the ranges below? Present the number of modulus operation(s) and sample  $m$  and  $n$  for the case.

$M = 89$ ,  $N = 144$ , 11 modulus

$1 \leq m \leq 150$   
 $1 \leq n \leq 150$

[**Hint:** Implement the Euclid algorithm in C++ or Java. After that, add the code to identify the number of modulus operation(s) for the numbers  $m$  and  $n$  in the ranges.]

=====

There are two programming problems in the homework.

This is the HackerRank link: <https://www.hackerrank.com/cst370-su20-hw1>

4. Write a C++ program (or Java program) called **hw1\_1.cpp (or hw1\_1.java)** that reads two groups of numbers in which each group has random integer numbers with possible duplicates. Your program should display the common numbers without any duplicates in the ascending order.

**Input format:** This is a sample input from a user.

CST370

5
7
20
8
7
15
3
8
14
7

The first number (= 5 in the example) indicates that there will be five numbers in the first group. Then, the numbers from the second line (= 7 in the example) to the sixth line (15 in the example) are actual numbers of the first group.

The following number (= 3 in the example) indicates that there are three numbers in the second group which are 8, 14, and 7.

For the input, your program should present the common numbers in both groups in the ascending order. In other words, your program should display **7** and **8** on the screen. If there's no common number, your program should display **NONE**.

**Sample Run 0:** Assume that the user typed the following lines

```
5
7
20
8
7
15
3
8
14
7
```

This is the correct output of your program.

Answer : 7 8

**Sample Run 1:** Assume that the user typed the following lines

```
2
-5
0
4
-2
-3
-4
5
```

This is the correct output of your program.

Answer : NONE

**Sample Run 2:** Assume that the user typed the following lines

```
5
2
```

```
-5
-5
2
7
4
2
7
7
-5
```

This is the correct output of your program.

Answer : -5 2 7

5. Write a C++ program (or Java program) called **hw1\_2.cpp (or hw1\_2.java)** that reads the time information of events in a convention center and displays the maximum number of concurrent events in the convention center.

**Input format:** This is a sample input from a user.

3
6 11
14 20
13 17

The first number (= 3 in the example) indicates that there are three event in the convention center. Then, the following lines are the time information (= start and ending hours) of each event. For example, the second line “6 11” indicates that there is an event which starts at 6:00(AM) and finishes at 11:00(AM). The next line indicates that there is an event from 2:00(PM) (= 14:00) to 8:00(PM) (= 20:00). The last event is from 1:00(PM) to 5:00(PM). In the homework, you should display the maximum number of events that can occur at the same time. For the input, your result should be 2 because the second and third events will happen at the same time from 2:00(PM) to 5:00(PM). For the homework, you can assume that the hour information always uses a positive integer from 0 to 23.

**Sample Run 0:** Assume that a user typed the following lines

```
3
6 11
14 20
13 17
```

This is the correct output.

Answer : 2

**Sample Run 1:** Assume that a user typed the following lines

```
4
10 15
10 12
9 10
8 9
```

This is the correct output. Note that there are three events at 10:00(AM).

Answer : 3

**Sample Run 2:** Assume that a user typed the following lines

```
4
10 12
2 9
13 17
18 23
```

This is the correct output.

Answer : 1

**Hint:** Pay attention to the starting and ending hours. Also, sorting and a counter will be useful.

**Note:** When you submit your homework program, don't forget to **include six items** such as "HackerRank link", "Title", "Abstract", "ID", "Name", and "Date".