HW-4: CSC241 (Data Structure and Algorithm)

Due Date: 12/01/2024 (11.59:59 PM)

**Special Instruction:** Your HW will be graded based on correctness and clarity. Keep you answer precise and to the point. If any question ask for justification of your answer/claim, you may receive a 0 if you merely provide an answer without justification. Your answer must need to be printed. Handwritten submissions will not be evaluated. All sub questions carry equal weights unless specified otherwise. Finally, please check the HW rules at the end

**What to do:** Reach to me ASAP if you have any confusion and/or have any emergency that may deter you to submit HW on time. Never hesitate to ask me if any of the previously discussed topics is unclear and you need some more discussion.

**What not to do:**

1. Ask to verify your solution
2. Ask to debug/analyze your code

***Submission instructions for all these Questions:*** *This is a programing assignment. Your answer script must contain the java code. Also, attach a screenshot from your output console/terminal.*

***Question 1:*** Open *restaurantHashTable.java*, which manages the order in a restaurant. The code contains the price for some items. Say, a customer, Nora, ordered 1 burger and 1 coke (say the order ID is 200). In that case the output should be:

Order ID: 200

Customer Name: Nora

Items: Burger (Quantity: 1), Coke (Quantity: 1)

Total Bill: $10.0

If an order ID (say order ID 210) is not found (or invalid), the output should be:

Order ID: 210

Order not found!

The code provided in “*restaurantHashTable.java*” is incomplete, and you need to write your code fill the following two methods.

*static double calculateBill(int orderId) {*

*//Complete the code*

*}*

*static void displayOrder(int orderId) {*

*//Complete the code*

*}*

The output should contain the Customer name, items ordered and total bill for OrderID 101, 102 and 103 (all these orders are created inside the *main()* method). The output format should follow the style as shown above.

A computer screen with white text

Description automatically generated

***Question 2:*** Given a matrix of dimension M \* N, where each cell in the matrix can have three different colors, **white, green, or grey,**which has the following meaning:

* White: Empty cell
* Green: Cells have fresh apples
* Grey: Cells have rotten apples

A rotten apple at index **(i,j)**can rot other fresh apples, which are its neighbors **(up, down, left, and right)**, and it takes 24 hours. Write a Java code that calculates the minimum required time, so all the apples become rotten.  If it is impossible to rot every apple, then return -1 and count the number of fresh apples.

For example, consider the following table (Initially):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

After 24 hours:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

After 48 hours:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

So it takes 48 hours to rot all the fresh apples.

You need to test your code and print the output for the following two tables:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

A screenshot of a computer

Description automatically generated

***Question 3:*** Write a Java code that calculates the largest rectangular area possible in each histogram where the largest rectangle can be made of many contiguous bars whose heights are given in an array. For simplicity, assume that all bars have the same width, and the width is 1 unit. For example, if the **Input** histogram is like this: {6, 2, 5, 4, 5, 1, 6} you can imagine an image like below.

A graph with numbers and a red rectangle

Description automatically generated

Here the answer is: 12 (covered by the red rectangle). Test your code for the following sample inputs:

**Input:** histogram = {3, 5, 1, 7, 5, 9}  
**Output:** Max area is ??

**Input:** histogram = {3, 5, 3, 7, 5, 9}  
**Output:** Max area is ??

A black background with white numbers and symbols

Description automatically generated

***Question 4:*** Given distinct integer numbers (sorted in ascending order) and **some** number ranges, write a java code that finds the number of points that lie in each of the ranges. A number will lie in a range **[A B]** if . Consider the following sample input:

Provide the numbers: 1 4 6 8 10

Range-1: 0 5

Range-2: 6 10

Range-3: 7 100

Numbers lie in Range 1: 2 (1 and 4 falls in this range)

Numbers lie in Range 2: 3 (6, 8 and 10 falls in this range)

Numbers lie in Range 3: 2 (8 and 10 falls in this range)

Test your code for the following sample inputs:

====================Input 1========================

Provide the numbers: 1 14 16 28 100 120

Range-1: 0 50

Range-2: 6 100

Range-3: 27 1000

Numbers lie in Range 1: ??

Numbers lie in Range 2: ??

Numbers lie in Range 3: ??

====================Input 2========================

Provide the numbers: 100 142 165 384 1000

Range-1: 0 50

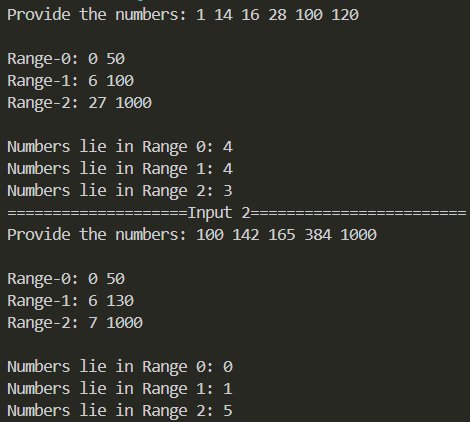
Range-2: 6 130

Range-3: 7 1000

Numbers lie in Range 1: ??

Numbers lie in Range 2: ??

Numbers lie in Range 3: ??



**Question 5:** There is a country with villages and each village have a number associated with them. There are also roads, and each of these M roads connect two different villages. One day, the president decided to test the hospitals to see if they can handle another wave of the coronavirus pandemic. In this country

* A village has sufficient vaccine if the number associated with it is a prime number. For example, villages numbered 2,3,5,7,11... will have sufficient vaccine with them. These villages can transfer vaccines to those villages who does not have vaccines.
* Any road that connects Village-x and Village-y, will take MAX(x,y) amount of time to travel this road. For example, if there is a road between Village-4 and Village-5, this road will take 5 time units to travel.
* If two villages are connected, there is only a single route.

For each village (in ascending order) find the minimum time required for the vaccine to arrive there. **If they already have sufficient vaccine, print 0. If it is impossible to transfer vaccine to a village, print -1 for that village.** Consider the following sample input:

Provide the number of villages: 6

Provide all the roads: 1-2, 1-3, 4-1, 5-2

Output: 2 0 0 6 0 -1

**Explanation:**

Village number 2,3 and 5 have their own vaccines so the time required for them is 0  
Village number 6 is not connected to any village which has a vaccine, therefore -1  
Village number 1,4 will get its vaccine in the shortest time from village 2

Test your code for the following sample inputs:

====================Input 1========================

Provide the number of villages: 9

Provide all the roads: 2-7, 3-4, 2-8, 1-6, 1-3, 2-3, 6-9, 4-5

Output: ??

====================Input 2========================

Provide the number of villages: 10

Provide all the roads: 1-4, 2-6, 5-8, 2-5, 1-2, 7-5, 3-8, 10-6

Output: ??

A screenshot of a computer program

Description automatically generated