**National University of Computer & Emerging Sciences, Karachi**
**Spring-2020 CS-Department**
**Final Examination**
**23rd Jun 2020, 09:00 AM – 12:30 PM**

| Course Code: CL218 | Course Name: Data Structures Lab |
|---|---|
| Instructor Name / Names: Ms. Farah Sadia, Ms. Ammara Yaseen, and Ms. Anam Qureshi | |
| Student Roll No: | Section No: |

Instructions:

**READ carefully the following instructions before attempting the paper.**

- The **Final Exam** consists of **four** questions. Make sure that you have all of these and that they are all legible.
- Read all questions and their instructions thoroughly before you begin. It is always worth your time to plan ahead!
- Write your **roll number** and **section** on each question documentation section.
- The Final Exam is worth 40 points. Each question is worth ten points.
- Points will be awarded based on your explicit answers. Partial credit will be given where possible, so show all of your work.
- Read the question properly, you can't find it on google so "don't waste your time". It's easy.
- You are supposed to make a program according to the question.
- Don't share your program, if your code is matched to any member of your class, both will get straight F in the course without asking who shared or who magically copied.
- You have to program it on dev c++. "Not ON Paper".
- In case of unavailability of internet due to load shedding etc, mobile data should be good enough to download and upload files. You do not need the internet during 3 hour offline exams. So better, fully charge your mobile battery and ready to upload data using mobile in case of emergency.
- **180 minutes** is for an exam to attempt, **30 minutes** for pdf formation and submission on LMS (Slate/ Google Classroom). If you find some problem with LMS, don't waste your time and email your lab instructor with subject: **FINAL_DS_SectionName.** Also submit the same later on LMS as well.
- Submissions after 12:30 pm are considered late. There is penalty for late submission i.e., deduction of 5 points for 10 minutes late submission.

**Time**: 180 minutes (to attempt) + 30 minutes (to submit) **Max Marks: 40**

## ***Question # 01:*** *Points: 10*

### ***BST:***

Implement a menu-driven program for managing a **"BUDGET BITES"** restaurant. All information about the available food is stored in a file. This information includes the **Food_ID**, **Deal_name**, **detail**, **served**, **quantity**, and the **price** of each item. When it is invoked, the program automatically creates a binary search tree with one node corresponding to one food item or deal and includes as its key the name of the item or deal. Another field in this node should include the position of the record in the file software. The only access to the information stored in software should be through this tree. The program should allow the file and tree to be updated when new deals arrive at the restaurant and when some deals are sold. The tree is updated in the usual way. All foods are entry ordered in the file software; if a new food arrives, then it is put at the end of the file. If the food already has an entry in the tree (and the file), then only the quantity field is updated. If a food item

or deal is sold out, the corresponding node is deleted from the tree (use deletion by copy technique), and the quantity field in the file is changed to 0. For example, if the file has these entries:

| Food_ID | Deal_name | Served | Quantity | Price |
|---|---|---|---|---|
| DEALS 1 | WHOPPER | Served with a small side of piping hot, thick cut French Fries or golden Onion Rings and a small fountain drink of your choice to make it a meal | 21 | 580 |
| DEALS 2 | SPICY CRISPY CHICKEN | Served with a small side of piping hot, thick cut French Fries or golden Onion Rings and a small fountain drink of your choice to make it a meal. | 10 | 650 |
| DEALS 3 | CHICKEN TIKKA BURGER | Served with a small side of piping hot, thick cut French Fries or golden Onion Rings and a small fountain drink of your choice to make it a meal. | 27 | 999 |
| SIDES 1 | NUGGETS 6 pcs | | 300 | 99 |
| SIDES 2 | Med FRENCH FRIES | Hot, thick cut Salted French Fries are golden on the outside and fluffy on the inside. | 150 | 199 |
| DESSERTS 1 | CARAMEL SUNDAE | Cool and creamy with a caramel swirl, our made-to-order Caramel Sundae is complemented by our velvety Vanilla dairy ice cream. | 66 | 245 |
| DESSERTS 2 | VANILLA SOFT SERVE CONE | One taste of our cool, creamy, and velvety Ice Cone, you'll think we perfected it. | 18 | 145 |

then after selling all 10 DEALS 2, the file is

| **Food_ID** | **Deal_name** | **Served** | **Quantity** | **Price** |
|---|---|---|---|---|
| DEALS 1 | WHOPPER | Served with a small side of piping hot, thick cut French Fries or golden Onion Rings and a small fountain drink of your choice to make it a meal | 21 | 580 |
| DEALS 2 | SPICY CRISPY CHICKEN | Served with a small side of piping hot, thick cut French Fries or golden Onion Rings and a small fountain drink of your choice to make it a meal. | 0 | 650 |
| DEALS 3 | CHICKEN TIKKA | Served with a small side of piping hot, thick cut French Fries or golden Onion Rings and a small fountain drink of | 27 | 999 |

| | | | | |
|---|---|---|---|---|
| | BURGER | your choice to make it a meal. | | |
| SIDES 1 | NUGGETS 6 pcs | | 300 | 99 |
| SIDES 2 | Med FRENCH FRIES | Hot, thick cut Salted French Fries are golden on the outside and fluffy on the inside. | 150 | 199 |
| DESSERTS 1 | CARAMEL SUNDAE | Cool and creamy with a caramel swirl, our made-to-order Caramel Sundae is complemented by our velvety Vanilla dairy ice cream. | 66 | 245 |
| DESSERTS 2 | VANILLA SOFT SERVE CONE | One taste of our cool, creamy, and velvety Ice Cone, you'll think we perfected it. | 18 | 145 |

If an exit option is chosen from the menu, the program cleans up the file by moving entries from the end of the file to the positions marked with 0 quantities. For example, the previous file becomes

| **Food_ID** | **Deal_name** | **Served** | **Quantity** | **Price** |
|---|---|---|---|---|
| DEALS 1 | WHOPPER | Served with a small side of piping hot, thick cut French Fries or golden Onion Rings and a small fountain drink of your choice to make it a meal | 21 | 580 |
| DEALS 3 | CHICKEN TIKKA BURGER | Served with a small side of piping hot, thick cut French Fries or golden Onion Rings and a small fountain drink of your choice to make it a meal. | 27 | 999 |
| SIDES 1 | NUGGETS 6 pcs | | 300 | 99 |
| SIDES 2 | Med FRENCH FRIES | Hot, thick cut Salted French Fries are golden on the outside and fluffy on the inside. | 150 | 199 |
| DESSERTS 1 | CARAMEL SUNDAE | Cool and creamy with a caramel swirl, our made-to-order Caramel Sundae is complemented by our velvety Vanilla dairy ice cream. | 66 | 245 |
| DESSERTS 2 | VANILLA SOFT SERVE CONE | One taste of our cool, creamy, and velvety Ice Cone, you'll think we perfected it. | 18 | 145 |

## ***Question # 02:*** *Points: 10*

### ***Stack:***

These are pancakes, a gaping hole in the "eat a healthy breakfast" story that Mom and Dad like to tell us. I'm not good enough at baking to make a perfect stack of pancakes all the same size, like the picture. Mine are more likely to look like the bottom stack. Still, they are yummy.
I like the order and symmetry of having my pancakes in order by size, largest on bottom. That way, the syrup and extra butter will run down and sweeten the whole stack! But my mom won't let me rearrange the pancakes by hand.

They, however, do let me use a spatula, aka **"the flipper"**. I am able to slip the spatula under any pancake and flip over the entire stack on top of the spatula. That took a lot of practice! I also have to rearrange the pancakes "in place" -- no extra plate allowed. So I do. (That took even more practice!)
But: how can I do this quickly, before my mom tells me to stop playing with my food? This is your task: help me put my pancakes in order.
*You have a stack of **n** pancakes of all **different sizes**, in no particular order. You are able to slip a spatula under any pancake and flip over the whole stack that is on the spatula. You would like to **arrange** my pancakes in order by size, largest on bottom.*
Let's proceed in two steps:

1. Just think for a minute about how you would do this.
2. Write down your code. How many times must you flip the pancakes?

Research paper: https://core.ac.uk/download/pdf/81955056.pdf

## ***Question # 03:*** *Points: 10*

### ***Sorting***

Mergesort() is used to merge the two sorted lists. An auxiliary array is required for merging operation. Moreover, the traditional merge sort algorithm does not take advantage of partial ordering of data. Your task is to implement merge sort algorithm which must consider the already sorted data. For example, in the array [5,6,7,2,3,0,10,11,12,1], subarrays with ordered elements are [5,6,7], [2,3], [0,10,11,12] and [1]. Therefore, first merge [5,6,7] and [2,3] to become [2,3,5,6,7], then subarrays [0,10,11,12] and [1] are merged to become [0,1,10,11,12], and finally subarrays [2,3,5,6,7] and [0,1,10,11,12] are merged to become [10,1,2,3,5,6,7,10,11,12].

## ***Question # 04:*** *Points: 10*

### *Hashing and Linked List*

Abdul Rehman is trying to memorize the vocabulary. Each time he randomly picks up a word from the dictionary. Memorize it and keep its record in a doc file. This record keeping helps him to avoid the confusions for selecting the same word again. For example, if he chooses the word "Empathy" and he can't recall if he has memorized this word earlier. Therefore, he will open the record and search for that particular word. If the word is present he will go for another word. This job of searching the word is quite hectic for Abdul Rehman as he has to look for the word one by one (manual searching).

Your task is to help him with the efficient solution of searching. Following is the list of words which he has already memorized.

1. Kindness
2. Politeness
3. Understandable
4. Empathy
5. Knowledge
6. Gratitude
7. Patience
8. Righteous
9. Appreciate
10. Easy

To do:

- Use the concept of hashing for efficient searching.
- Use the following hash function to perform the operations. If the word is "Thought"
    - Break the word in characters.
    - Use ASCII of each character and add the ASCII values to get a number format.
    - For "Thought", the ASCII values are : 84+104+111+117+103+104+116
    - At the end also add sum of digits of your roll number, for example if your roll number is 18k-1001. It will be considered as 1+8+1+0+0+1=11
        - Add 11 in the ASCII values of the word
        - 84+104+111+117+103+104+116+11= 750
    - Now use this number as a key in hashfunction
    - Hashfunction = Key mod size_of_table
    - Size of table is 20
- In case of collisions, use separate chaining method only.
- Also count the number of nodes on each index.
- Show the final hash table

Note: Cheating is a crime; try to do on your own!
Good Luck ☺