



COMSATS UNIVERSITY ISLAMABAD (CUI)

DEPARTMENT OF COMPUTER SCIENCE
MIDTERM EXAMINATION SPRING - 2024
BS (AI) III SEMESTER

SP23-BCS-098
M. Anis

Course: CSC211-Data Structures & Algorithms

Maximum Marks: 25

Dated:

Instructor: Dr. Inayat Ur-Rehman

Time Allowed: 80 Minutes

- All questions are self-explanatory and require no further explanations during exam time.
- Make sure that you have signed the attendance sheet before leaving the examination room.
- Return the question paper along with the answer sheet.
- Attempt all questions.

(CLO – 1: Employ linear data structures to solve computing problems.)

Q.# 1. Consider an unordered singly linked list of integers. Write a C/C++ function to split and display it into two halves (resulting into two linked lists) such that both new linked lists are sorted in ascending order. Following is an instance of input-output for further clarification. [5 Marks]

Input: 12->56->2->11->24->45->38->29

Output: list1 is 2->11->12->56 and list2 is 24->29->38->45

Q.# 2. Implement a basic calculator in C/C++ to evaluate a postfix expression of strings using stack ADT. The expression string may contain multiply *, divide /, plus +, minus - signs and non-negative integers (single digits but can result into two or three digit numbers during evaluation). Do not redefine stack ADT just write evaluate() function. Following is an instance of input-output for further clarification. [6 Marks]

Input: 23-4+567*+*

Output: 141

Q.# 3. Suppose you have a double ended queue where you can en-queue a new element at front or rear and you can also de-queue an element from front or rear end of it. In order to maximize the performance of a double ended queue write a circular version of it in C/C++ (just define a structure and write enqueue() function) to incorporate the following situation. [7 Marks]

0	1	2	3

front = -1 rear = -1

14			
0	1	2	3

Insert at rear

front = 0 rear = 0

14	25		
0	1	2	3

Insert at rear

front = 0 rear = 1

14	25		53
0	1	2	3

Insert at front

front = 3 rear = 1

(CLO – 2: Use non-linear data structures to solve computing problems.)

Q.# 4. Pakistan International Airport is developing a computer simulation of air-traffic control, which handles events such as landings and take-offs. Each event can be defined by an eventId(string) and a timeStamp (int). Time stamp denotes when the event occurs. The simulator needs to perform efficiently the following two operations: **[7 Marks]**

- a. Insert a new event with a given timeStamp (i.e. add a future event)
- b. Get the event with the smallest timeStamp (i.e. determine the next event to process)

4 Which data structure would you use to support the above operations so that **without applying any explicit sorting process each event must go to its sorted position in the chosen data structure?** Write a C/C++ code for your provided solution. Keep in your mind that number of flights is not fixed and application is time sensitive especially with respect to finding an event in the given search space.

*******BEST OF LUCK*******