



FAST- National University of Computer & Emerging Sciences, Karachi.
School of Computing,
Mid Term Examination, Spring 2023
17th-March, 2023, 8:30 am – 10:30 am



Course Code: CL-2001	Course Name: Data Structures Lab
Instructors: Safia Baloch	
Student Roll No:	Section:

Instructions:

- Except your Roll No and Section, DO NOT WRITE anything on this paper.
- Return the question paper with your answer sheet.
- Read each question completely before answering it. There are **3 questions on 2 pages**.
- Create a folder on the desktop with your roll number. Save each question in the folder as 1.cpp, 2.cpp, and 3.cpp.
- Zip the folder and submit it to the folder of (Data Structures Lab Exam) on the local network.

Time Allowed: 120 minutes

Maximum Points: 25

Arrays and Sorting	
Question No. 1	[Time: 45 minutes] [Points: 9]

A student has a collection of academic books, each with a unique title that begins with a different letter of the alphabet. The student needs to store these books in a way that allows them to be sorted in ascending order using the Shell sort algorithm. To accomplish this, the student has decided to create a program in C++ that uses a dynamic array to store the book titles. The program will prompt the user to enter the number of books they want to store, and then ask for the title of each book, one at a time. To ensure that the program is well-designed and efficient, the student plans to implement the rule of three in their C++ program. This means that they will define a copy constructor, a copy assignment operator, and a destructor for any class that manages dynamic memory. By following this best practice, the student can ensure that their program behaves correctly and avoids memory leaks or other issues.

Linked List	
Question No. 2	[Time: 30 minutes] [Points: 7]

Suppose you have a doubly circular linked list that represents a playlist of songs, where each node represents a song and contains the song's name and artist name. You want to allow the user to remove a song at a specific position in the playlist. The user will enter the name, artist name as well as the position from where it should be removed. Write a function in C++ that takes the head pointer of the linked list, the name, artist name, and removes that song at the specified position in the playlist. If the position is out of range (less than 1 or greater than the length of the playlist), the function should return without modifying the linked list.

<pre> #include <iostream> using namespace std; struct Node { string song; Node *prev,*next; }; class Playlist { private: Node *head; Node *tail; int count; public: Playlist() { // Constructor head = NULL; tail = NULL; count = 0; } ~Playlist() { // Destructor Node *temp = head; while (temp != NULL) { Node *current = temp; temp = temp->next; delete current; } head = NULL; tail = NULL; count = 0; } </pre>	<pre> // Function to remove a song from the playlist at a given position void removeSong(int position) { Paper B WRITE THIS FUNCTION } // Function to display the playlist void display() { if (head == NULL) { cout << "List is empty." << endl; return; } Node* temp = head; do { cout << temp->song << " "; temp = temp->next; } while (temp != head); cout << endl; } int main() { Playlist myList; myList.display(); myList.removeSong(3); myList.display(); return 0; } </pre>
---	---

Recursion and Backtracking		
Question No. 3	[Time: 45 minutes]	[Points:9]

Write a program that takes an integer n and prints all $n!$ permutations of the n letters starting at a (assume that n is no greater than 26). A permutation of n elements is one of the $n!$ possible orderings of the elements. As an example, when $n = 3$ you should get the following output (but do not worry about the order in which you enumerate them):

$n=2$

ab ba

$n=3$

bca cba cab acb bac abc

***** Good Luck *****