Assignment I

Arrays, Linked lists

Submission: September 9th 2024.

Late submissions will get 50% penalty. Last date with penalty 11th September.

Submissions after 11th September will get zero.

Plagiarism will be checked

Arrays

Q1. Temperature data is stored as a struct. The elements are city, date and array of temperature readings taken over the day. Dynamically create an array of temperature data. Write a menu driven program to perform the following operations:

a. Write a function to insert a new element at any user given position of the array. Test your program by inserting an element
   i. At the beginning of the array
   ii. At the end of the array
   iii. after the second element
   iv. before the third element
b. Write a function to search for a user given city in the array
c. Write a function to print the entire array.
d. Write a function to print the entire array but in reverse direction
e. Write a function to delete an element from any position in the array. There should be no empty slot in the array. Resize your array using dynamic memory allocation. Test your program by deleting from
   i. the beginning of the list
   ii. the end of the list
   iii. after the second element

Q2. You are given two sorted arrays arr1 and arr2 of integers. Merge the two arrays into one sorted array. Handle all the test cases.

Single Linked lists

Q3. Assume that there is a sequence of song tracks that has to be played. Create a linked list of songs where each song is a node with song id and song's name as the data part of the node. Write a menu driven program to perform the following operations:

a. Write a function to insert a new song at the beginning of the list
b. Write a function to insert a new song at the end of the list
c. Write a function to insert a new song **after** the node with song id = x. Take x from the user.
d. Write a function to insert a new song **before** the node with song id = x. Take x from the user.
e. Write a function to print the entire linked list
f. Write a function to search for a song in the linked list (assume the list is not sorted). The name of the song is to be an input from the user
g. Write a recursive function to print the entire linked list but in reverse direction

h. Write a function to *count* the number of elements of a linked list.
i. Convert the above function, *count*, to a recursive function.
j. Write a function to delete the song with song id = x. Take x from the user.
k. Write a function to delete the **last** song of the linked list
l. Write a function to delete the **first** song of the linked list
m. Write a function to delete ALL the nodes of the linked list using the:
    i. Iterative approach
    ii. Recursive approach

Q4. There are several episodes for a web series. The details of each episode are stored as a node in a linked list. The data of each node contains serial id, episode number, array of customer numbers. Create a linked list for your favourite web series and add 8 episodes to it. Next, modify the list from some position, n, such that the episode at position n is the first node of the linked list. For this, write two functions:
    a. Rotate the list from n, in the clockwise direction
    b. Rotate the list from n, in the anti-clockwise direction

Q5. Read a very very long integer as a string (greater than the size of long integer) and then convert each character digit into a number and then store it in a linked list. Split this linked list into two sub-lists as follows: The first sub-list consists of the odd integers, and the second sub-list the even integers of the original list

Q6. Given a linked list of non-negative integers, write a function to swap every two adjacent nodes and return its head pointer. Note: Do not swap by exchanging the values of the nodes.

Q7. You are given two sorted linked lists list1 and list2 of non-negative integers. Merge the two lists into one sorted list.