CSN-261: Data Structures Laboratory

Lab Assignment 2 (L2)

Instructions:

1) **Use either C/C++ for solving the assignment.**
2) **Throughout the assignment, *n* represents the number of input.**
3) **Array index starts with 0 in C++.**
4) **RED color indicates the input in each test case.**

Problem 1. Write a program to find and display the k^{th} smallest element in an array.

Test Case:

Input: Print: Enter the size of Array : **8**
Print: Enter the elements : **909 967 552 524 735 383 616 718**
Print: Enter the Kth smallest you want to find: **3**
Print: kth smallest element : **552**

Problem 2. Create a memory and time efficient linked list data structure to maintain a database of *n* number of newly enrolled students which can store the following information.

a. Roll number/Mobile number/other unique ID
b. Full Name
c. Course Code
d. Age (only in integer value)
e. Branch

The final students linked list should be **SORTED** following the age.

Test Cases: **Print:** Enter the number of students: **2**

Print: Enter the choice for student 1
R for Roll number
M for Mobile number
O other unique ID : **M**

Print: Enter 10-digits Mobile number of the student 1: **9995343124**
Print: Enter the full name for student 1: **Ramesh Kuamr**
Print: Enter the course code for student 1: **CSN-261**
Print: Enter the age for student 1: **19**
Print: Enter the branch name for student 1: **CSE**
Print: Enter the choice for student 2
R for Roll number
M for Mobile number
O other unique ID : **R**

Print: Enter the Roll number of student 2: **CSE202001**
Print: Enter the full name for student 2: **Hemant Gupta**
Print: Enter the course code for student 2: **CSN-520**
Print: Enter the age for student 2: **18**
Print: Enter the branch name for student 2: **EC**

The sorted list of students is :
1. CSE202001, Hemant Gupta, CSN-520, 18, EC
1. 9995343124, Ramesh Kuamr, CSN-261, 19, CSE

Problem 3. Create a memory and time efficient Stack data structure to store heterogeneous (Int/Char/Float) elements. Write Push (), Pop(), Full (), Empty () etc. as the user defined functions to insert and delete the items.

Test Cases: **Print:** Enter your choice

I for Insert
D for Delete
E for exit
P for Print : **I**

Print: Enter **I** for integer, **C** for char and **F** for float: **I**
Print: Enter an integer value: **5**
Print: Enter your choice

I for Insert
D for Delete
E for exit
P for Print : **I**

Print: Enter **I** for integer, **C** for char and **F** for float: **F**
Print: Enter a Float value: **7.59**
Print: Enter your choice

I for Insert
D for Delete
E for exit
P for Print : **I**

Print: Enter **I** for integer, **C** for char and **F** for float: **C**
Print: Enter a Char value: **A**
Print: Enter your choice

I for Insert
D for Delete
E for exit
P for Print : **D**

Print: The popped eminent is : A
Print: Enter your choice

I for Insert
D for Delete
E for exit
P for Print : **P**

Print : The Current status of the Stack is: 7.59
5

Print: Enter your choice

I for Insert
D for Delete
E for exit
P for Print : **E**

Print: Program is Stopped.

Problem 4. Implement a memory and time efficient **Circular Queue** data structure for heterogeneous (Int/Char/Float) elements using linked list. Write user defined operations such as Overflow (), Underflow (), Insert (), Delete () etc.

Test Cases: **Print:** Enter your choice

I for Insert
D for Delete
E for exit
P for Print : **I**

Print: Enter **I** for integer, **C** for char and **F** for float: **I**
Print: Enter an integer value: **5**
Print: Enter your choice

I for Insert
D for Delete
E for exit
P for Print : **I**

Print: Enter **I** for integer, **C** for char and **F** for float: **F**
Print: Enter a Float value: **7.59**
Print: Enter your choice

I for Insert
D for Delete
E for exit
P for Print : **I**

Print: Enter **I** for integer, **C** for char and **F** for float: **C**
Print: Enter a Char value: **A**
Print: Enter your choice

I for Insert
D for Delete
E for exit
P for Print : **D**

Print: The Deleted eminent is : 5
Print: Enter your choice

I for Insert
D for Delete
E for exit
P for Print : **P**

Print : The Current status of the Queue is: 7.59
A
Print: Enter your choice

I for Insert
D for Delete
E for exit
P for Print : **E**

Print: Program is Stopped.