


National University of Computer and Emerging Sciences, Lahore Campus

	Course: Program: Name: Registration #: Due Date:	Data Structure BSCS 11th March 2020	Course Code: Semester: Section: Assessment	4th 4A, 4B Assignment 1
---	---	---	---	--

Instruction/

Notes:

1. Late submissions will not be entertained.

Q1:

Design a method for keeping two stacks within a single linear array *stack[max_size]*, so that neither stack overflows until all of memory is used and an entire stack is never shifted to a different location within the array. Write C++ routines, push1, push2, pop1, pop2 to manipulate the two stacks.

Q2:

The FAST PARKING GARAGE contains a single lane that can hold up to 10 cars. There is only a single entrance/exit to the garage at one end of the lane. If a customer arrives to pickup a car that is not nearest to the exit, all cars blocking the path are moved out, the customer's car is driven out, and the other cars are restored in the same order that they were in originally. Write a **program** that processes a group of input lines. Each input line contains an 'A' for arrival and 'D' for departure, and a license plate number. Cars are assumed to arrive and depart in the order specified by the input. The program should print a message whenever a car arrives and departs. When a car arrives, the message should specify whether or not there is a room for the car in the garage. If there is no room, the car leaves without entering the garage. When a car departs, the message should include the number of times that the car was moved out of the garage, to allow other cars to depart.

Q3:

Assume a machine that has a single register and six instructions

Instruction	Variable	Description
-------------	----------	-------------

s

LD	A	Places the operand A into the register
ST	A	Places the contents of the register into the variable A
AD	A	Adds the contents of the variable A to the register
SB	A	Subtracts the contents of the variable A from the register

ML	A	Multiplies the contents of the register by the variable A
-----------	---	---

DV	A	Divides the contents of the register by the variable A
-----------	---	--

- a. Write a program in C++ that accepts a postfix expression containing single-letter operands and the operators +, -, *, and /, and prints a sequence of the instructions to evaluate the expression and leave the result in the register.

Use variables of the form AUX_n as temporary variables. For example, using the post expression **ABC*+DE-/** should print the following:

```
LD    B
ML    C
ST    AUX1
LD    A
AD    AUX1
ST    AUX2
LD    D
SB    E
ST    AUX3
LD    AUX2
DV    AUX3
ST    AUX4
```

Q4:

Double ended queue, deque (usually pronounced like "deck") is an irregular acronym of double-ended queue. Double-ended queues are sequence containers with dynamic sizes that can be expanded or contracted on both ends (either its right or its left). Call two ends of the deque as left and right. How can a deque be represented as a data structure in C++.

- a. Write four C++ routines

remLeft, remRight, insertLeft, insertRight

to remove and insert elements to the right and left ends of the deque. Make sure that the routines work properly for the empty deque and underflow and overflow conditions are properly implemented.

- b.** Define an *input-restricted deque* as a deque for which only the operations remLeft, remRight, and insertRight are valid.
- c.** Define an *output-restricted deque* as a deque for which only the operations remLeft, insertLeft, and insertRight are valid.

Q5:

A **biotonic** doubly linked list is a doubly linked list which is first increasing and then decreasing. A strictly increasing or a strictly decreasing list is also a biotonic doubly linked list. For example, the given list is an example of **biotonic** doubly linked list.

2<-> 5 <-> 7 <-> 12 <-> 10 <-> 6 <-> 4 <-> 1

Write down a code in C++ to sort the biotonic list. The algorithm must be as efficient as possible.