

National Institute of Technology, Calicut
Department of Computer Science and Engineering
Monsoon2021
CS2092D – PROGRAMMING LABORATORY (MCA-I A)
Assignment-7

Policies for Submission and Evaluation

You must submit your assignment on the moodle (Eduserver) course page on or before the submission deadline. Also, ensure that your programs in the assignment must compile and execute without errors in Athena server. During the evaluation your uploaded programs will be checked in Athena server only. Failure to execute programs in the assignment without compilation errors may lead to zero marks for that program.

Your submission will also be tested for plagiarism by automated tools. In case your code fails to pass the test, you will be straightaway awarded zero marks for this assignment and considered by the examiner for awarding F grade in the course. Detection of ANY malpractice regarding the lab course will also lead to awarding an F grade.

Naming Conventions for Submission

Submit a single ZIP (.zip) file (do not submit in any other archived formats like .rar or .tar.gz). The name of this file must be ASSG<NUMBER>_<ROLLNO>_<FIRSTNAME>.zip. (For example: ASSG4_BxxxxxyCS_LAXMAN.zip). DO NOT add any other files (like temporary files, inputfiles, etc.) except your source code, into the zip archive. The source codes must be named as

ASSG<NUMBER>_<ROLLNO>_<FIRSTNAME>_<PROGRAM-NUMBER>.<extension>

(For example: ASSG4_BxxxxxyCS_LAXMAN_1.c). If there are multiple parts for a particular question, then name the source files for each part separately as in

ASSG4_BxxxxxyCS_LAXMAN_1b.c.

If you do not conform to the above naming conventions, your submission might not be recognized by some automated tools, and hence will lead to a score of 0 for the submission. So, make sure that you follow the naming conventions.

Standard of Conduct

Violations of academic integrity will be severely penalized. Each student is expected to adhere to high standards of ethical conduct, especially those related to cheating and plagiarism. Any submitted work MUST BE an individual effort. Any academic dishonesty will result in zero marks in the corresponding exam or evaluation and will be reported to the department council for

record keeping and for permission to assign an F grade in the course. The department policy on academic integrity can be found at:

http://minerva.nitc.ac.in/cse/sites/default/files/attachments/news/Academic-Integrity_new.pdf .

Assignment 7 **Questions**

1.

(a) Implement a Queue using Linked List

A Query T is of 2 Types

(i) 1 *num* (a query of this type means enqueue '*num*' into the queue)

(ii) 2 (a query of this type means to dequeue an element from the queue and print that element)

Sample input and output with explanation

Input:

T = 5 (No. of queries)

1 2 1 3 2 1 4 2

1. 1 2 – Enqueue 2 – The queue will be {2}

2. 1 3 – Enqueue 3 – The queue will be {2 3}

3. 2 – Dequeue – Print 2 and the queue will be {3}

4. 1 4 – Enqueue 4 – The queue will be {3 4}

5. 2 – Dequeue – Print 3 and the queue will be {4}

Output:

2 3

Test cases

Test case 1	Test case 2
Input: T = 3 1 7 1 9 2	Input: T = 1 2
Output: 7	Output: No element in the queue

(b) Implement a queue using two Stacks, s1 and s2.

A Query Q is of 2 Types

(i) 1 x (a query of this type means enqueue 'x' into the queue)

(ii) 2 (a query of this type means to dequeue an element from queue and print that element)

Note: Your program should be designed so that the dequeue operation takes $O(N)$ time, and enqueue will take $O(1)$ time in terms of time complexity.

Sample input and output with explanation

Input:

5 (No. of queries)

1 7 1 6 2 1 4 2

1. 1 7 – Enqueue 7 – The queue will be {7}
2. 1 6 – Enqueue 6 – The queue will be {7 6}
3. 2 – Dequeue – Print 7 and the queue will be {6}
4. 1 4 – Enqueue 4 – The queue will be {6 4}
5. 2 – Dequeue – Print 6 and the queue will be {4}

Output:

7 6

Test cases

Test case 1	Test case 2
Input: 4 1 1 1 3 1 6 2 Output: 1	Input: 3 1 4 2 2 Output: 4 -1 <i>(Note: If the queue is empty, then print -1)</i>

2.

(a) Akhil owns a collection of N disks. Akhil likes to make his disks in stacks with a fundamental rule - the disks in a single stack must be sorted by radius in a strictly non-decreasing manner, with the lowest radius at the top. You need to write a function that takes a stack and make another stack that has all disks in sorted order according to their radius.

For example, a stack of disks with radii (5, 2, 1) is valid, while a stack of disks with radii (3, 4, 1) is not

Note: Complete the task using Stack implementation of Link-list

Test cases

Test case 1	Test case 2
Input: [23 313 39 101 9 58]	Input: [9 8 7 6 5]
Output: [9 23 39 58 101 313]	Output: [5 6 7 8 9]

(b) Dholakpur's kids have again started stealing TunTun aunty's ladoos. As TunTun Aunty is cautious now, she wants to hide these ladoos in a stack. The stack can have the ladoos on top of each other only. The ladoos are of different sizes, starting from 1, and initially, the stack is empty. You are given an array of integers denoting the size of the ladoos. If the size of the ladoo is greater than 0, then insert the ladoo on the top of the stack. If it is 0, you can eat the topmost ladoo in the stack. It is guaranteed that the same ladoo won't be added more than once. After inserting all ladoos into the stack, you need to check whether some given pair of ladoo is present together or not. So you need to write a function that takes a stack as input return " Yes" if required pair is present together else return " No".

Note: Complete the task using Stack implementation of Array. Also, while inputting ladoos into the stack, ensure that zero-sized ladoos are not placed first.

Test cases

Input:N=6, arr=[1 3 0 0 4 5], pair=[1,4]

Output: No

Input: N=6, arr=[1 3 0 0 4 5], pair=[4,5]

Output: Yes

Input: N=10, arr=[1 3 0 0 4 5 7 0 9 2], pair=[7,9]

Output: No

Input: N=10, arr=[1 3 0 0 4 5 7 0 9 2], pair=[5,9]

Output: Yes
