



राष्ट्रीय प्रौद्योगिकी संस्थान जमशेदपुर NATIONAL INSTITUTE OF TECHNOLOGY JAMSHEDPUR

(An Institution of National Importance under MHRD, Government of India)

Department of Computer Science and Engineering

Assignment 2: Stack and Queue

Programme Name: B.Tech.		Semester: 3 rd
Course Code: CS1304	Course Name: Data Structure	
Branch: B.Tech		

Instructions

1. Write the assignment in your own handwriting, scan it and upload it on to the google classroom.

Q No. Question Description

Q 1 Suppose that you have an $m \times n$ maze of rooms. Each adjacent pair of rooms has a door that allows passage between the rooms. At some point of time some of the doors are locked, the rest are open. A mouse sits at room number (s,t) and there is fabulous food for the mouse at room number (u,v) . Your task is to determine whether there exists a route for the mouse from room (s,t) to room (u,v) through the open doors. The idea is to start a search at room no (s,t) , then investigate rooms $(s_1,t_1), \dots, (s_k,t_k)$ that can be reached from (s,t) and then those rooms that can be reached from each (s_i,t_i) , and so on. There is no need to revisit a room during the search. Maintain an $m \times n$ array of flags in order to keep track of the rooms that are visited.

Use a stack to implement the search. Initially push (s,t) to the empty stack. Subsequently, as long as the stack is not empty, consider the room (x,y) at the top of the stack. If (x,y) has a yet unvisited neighboring room, push that room at the top of the stack. If (x,y) does not have an unvisited neighboring room, pop (x,y) out of the stack. If during these operations, the desired room (u,v) ever appears at (the top of) the stack, then a route from (s,t) to (u,v) is detected. If the search completes (i.e., the stack becomes empty) without ever having (u,v) in the stack, then there is no $(s,t)-(u,v)$ path.

Q2 Implement the above search (**in Q1**) using a queue. Maintain a queue of rooms to search from. Initially enqueue (s,t) to an empty queue. Subsequently, as long as the queue is not empty, look at the room (x,y) at the front of the queue. If $(x,y) = (u,v)$, then report success and return. Else dequeue (x,y) from the front and enqueue all unvisited neighboring rooms at the back of the queue. If the search stops (i.e., the queue becomes empty) without ever having (u,v) at the front of the queue, report failure.

Q 3 Write a routine *prefix* to accept an infix string and create the prefix form of that string. Assuming that the string is read from right to left and that the prefix string is created from right to left.

Q 4 Can a template be used to store elements of different types on the same stack? Why or why not?

Q 5

The Bashemin Parking Garage contains a single lane that holds up to ten cars. There is only a single entrance/exit to the garage at one end of the lane. If a customer arrives to pick up a car that is not nearest the exit, all cars blocking its 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 or a '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 or departs. When a car arrives, the message should specify whether or not there is 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.