**ABHISHEK SHARMA**

**CS 2nd Year**

**Section : “I”**

**Roll no.: 01**

**Enrollment No.: 12019009001127**

**DATA STRUCTURE AND ALGORITHM LAB**

**WEEK : 12**

**HackerRank ID : 12019009001127\_I**

**Date : 03.11.2020**

**Q1. Queue using two stacks problem.**

**Code :**

#include <stack>

#include <cmath>

#include <cstdio>

#include <vector>

#include <iostream>

#include <algorithm>

using namespace std;

int main()

{

stack<int> front, rear ;

int q, n, x ;

cin >> q ;

while(q--)

{

cin >> n ;

if(n == 1)

{

cin >> x ;

rear.push(x) ;

}

else

{

if(front.empty())

{

while(!rear.empty())

{

front.push(rear.top()) ;

rear.pop() ;

}

}

if(!front.empty())

{

if(n == 2)

{

front.pop() ;

}

else if(n == 3)

{

cout << front.top() << endl ;

}

}

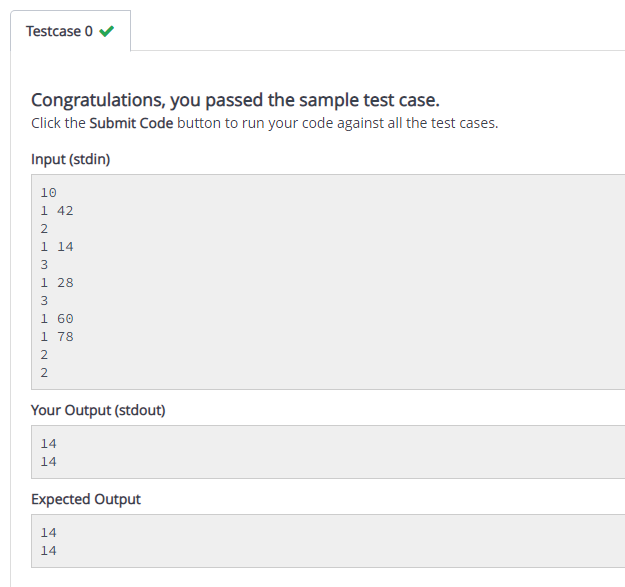
}

}

return 0;

}

**Output :**

****

**Q2. Write a program to calculate the number of items in a queue implemented through Linked List.**

**Code :**

l = []

i = int(input()) #enter the query

for a in range (0,i):

a = input()

if (a[0]=='1'):

l.append(a[-1])

elif (a[0]=='3'):

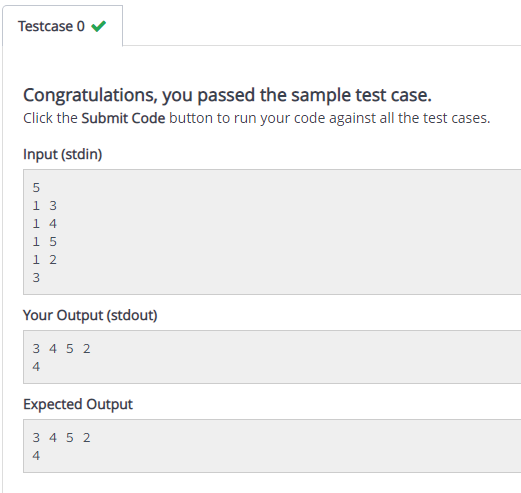
for b in range (0,i-1):

print (l[b], end = " ")

print ()

print (len(l))

**Output :**

****

**Q3. Write a program to create a queue using array which permits insertion at both the ends. The options provided are shown below:**

**Code :**

l = []

i = int(input()) #enter the query

for a in range (0,i):

a = input()

if (a[0]=='1'):

l.append(a[-1])

elif (a[0]=='2'):

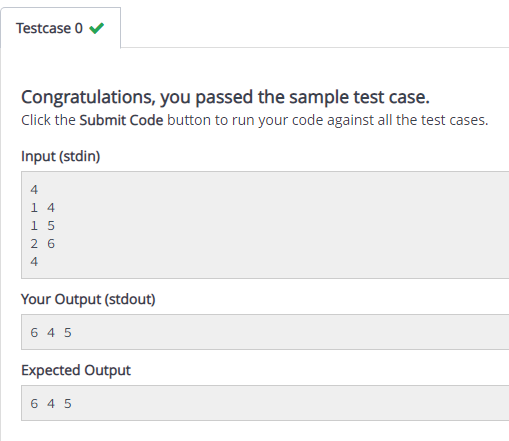
l.insert(0,a[-1])

elif (a[0]=='4'):

for b in range (0,i-1):

print (l[b], end = " ")

**Output :**

****

**Q4. Write a program to implement a dequeue with the help of a linked list.**

**Code :**

l = []

i = int(input()) #enter the query

for a in range (0,i):

a = input()

if (a[0]=='1'):

l.append(a[-1])

elif (a[0]=='2'):

l.insert(0,a[-1])

elif (a[0]=='5'):

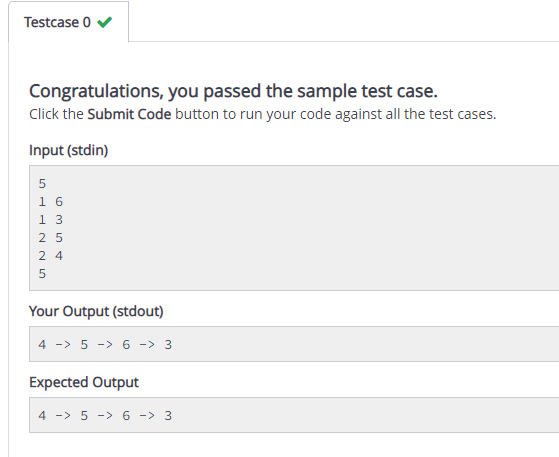
for b in range (0,i-1):

print (l[b], end = " ")

if (b<3):

print ("->", end = " ")

**Output :**

****

**Q5. Write a program to implement a priority queue using Linked List.**

**Code :**

def enqueue (x,y):

global k

if (int(y)>int(k)):

l.append(x)

k = y

elif (int(y)<int(k)):

l.insert(0,x)

k = y

def display():

for j in range (0,len(l)):

print ('{}|{}'.format(l[j],j+1),end = " ")

if (j<(len(l)-1)):

print ("->",end = " ")

i = 0

l = []

k = 0

count = int(input())

while (i<count):

a = input()

if (a[0]=='1'):

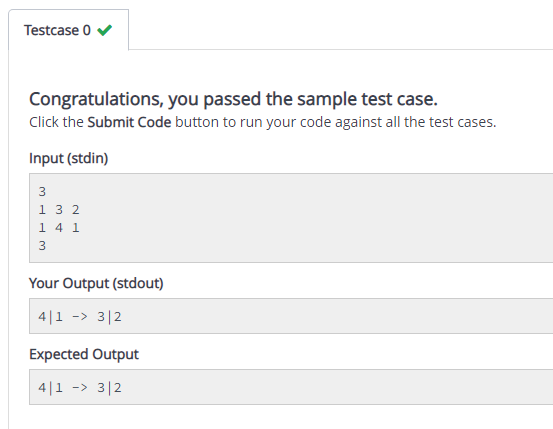
enqueue (a[2],a[4])

elif (a[0]=='3'):

display()

i = i+1

**Output :**

****

**Q6. Write a program to input two queues (A and B) and compare their contents.**

**Code :**

l1= []

l2 = []

for a in range (0,11):

a = input()

if (a[0]=='1'):

l1.append(a[-1])

elif (a[0]=='2'):

l2.append(a[-1])

elif (a[0] == '6'):

for b in range (0,len(l1)):

print (l1[b], end = "")

print()

elif (a[0] == '7'):

for b in range (0,len(l2)):

print (l2[b], end = "")

print()

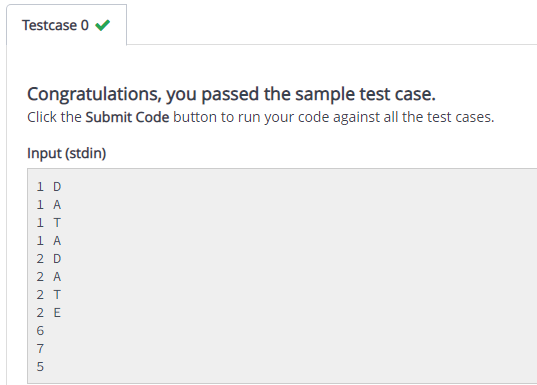
for j in range (0, len(l1)):

if (l1[j]!=l2[j]):

print ("Not Identical")

print ("Mismatch at {}th element".format(j+1))

**Output :**

****

****

**Q7. Write a program to implement Multiple queue where an array Queue[n] is used to represent two queues, Queue A and Queue B.**

**Code :**

l1= []

l2 = []

i = int(input())

for a in range (0,i):

a = input()

if (a[0]=='1'):

l1.append(a[-1])

elif (a[0]=='2'):

l2.append(a[-1])

elif (a[0] == '5'):

for b in range (0,len(l1)):

print (l1[b], end = " ")

print()

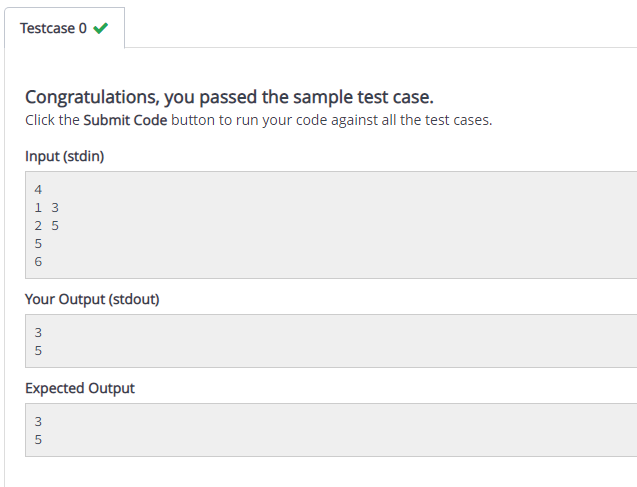
elif (a[0] == '6'):

for b in range (0,len(l2)):

print (l2[b], end = " ")

print()

**Output :**

****

**Q8. A palindrome is a word, phrase, number, or other sequence of characters which reads the same backwards and forwards. Can you determine if a given string *s*, is a palindrome?**

**Code :**

def isPalindrome(s):

return (s == s[::-1])

s = input()

ans = isPalindrome(s)

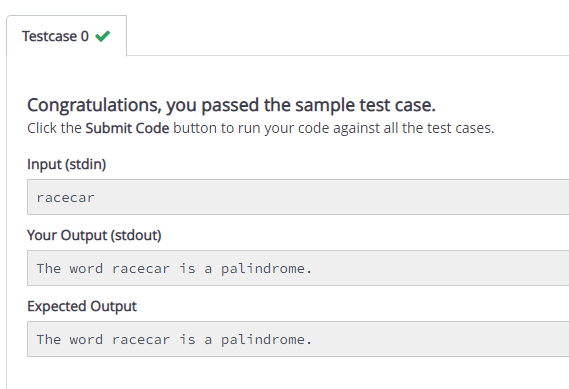
if ans:

print("The word {} is a palindrome.".format(s))

else:

print("The word {} is not a palindrome.".format(s))

**Output :**

****