**Godavari College Of Engineering, Jalgaon.**

**Subject Name:** Data Structure **Teacher Name:** Prof.S.S.Shete

**Practical No**. : 7 **Date:**

**Class: S**.E **Roll No:**

**Title:** Write a program to implement stack using two Queue such that push operation runs in linear time and pop operation runs in constant time.

**Theory:**

This is a C Program to implement stack using queue. The idea is pretty simple. We start with an empty queue. For the push operation we simply insert the value to be pushed into the queue. The pop operation needs some manipulation. When we need to pop from the stack (simulated with a queue), first we get the number of elements in the queue, say n, and remove (n-1) elements from the queue and keep on inserting in the queue one by one. That is, we remove the front element from the queue, and immediately insert into the queue in the rear, then we remove the front element from the queue and then immediately insert into the rear, thus we continue up to (n-1) elements. Then we will perform a remove operation, which will actually remove the nth element of the original state of the queue, and return.

# Algorithm:

1. When calling the **push** function, simply **enqueue** the elements into the **queue 1**.
2. If it call **pop** function
3. **Deque** all the elements from **queue 1** and enqueue into **queue 2** except the recently enqueued element of **queue 1.**
4. Now **deque** recently inserted element from **queue 1** and **display** it.
5. **Deque** all elements of **queue 2** and **enqueue** into **queue 1**.

**Program:**

#include<stdio.h>

#include<stdlib.h>

#define maxsize 5

int rear1=-1, front1=-1, queue1[maxsize];

int rear2=-1, front2=-1, queue2[maxsize];

void insertion1(int ele)

{

rear1=rear1+1;

queue1[rear1]=ele;

if(front1==-1)

{

front1=0;

}

}

int deletion1()

{

int x; x=queue1[front1];

if(front1==rear1)

{

front1=rear1=-1;

}

else

{

front1++;

}

return x;

}

void insertion2(int ele)

{

rear2++;

queue2[rear2]=ele;

if(front2==-1)

{

front2=0;

}

}

int deletion2()

{

int x;

x=queue2[front2];

if(front2==rear2)

{

front2=rear2=-1;

}

else

{

front2++;

}

return x;

}

void push()

{

int ele;

printf("Enter element to be insert: "); scanf("%d",&ele);

insertion1(ele);

}

void pop()

{

int x;

while(front1!=rear1)

{

x=deletion1();

insertion2(x);

}

x=deletion1();

printf("%d",x);

while(front2!=-1)

{

x=deletion2(); insertion1(x);

}

}

void display()

{

int i;

for(i=front1;i<=rear1;i++)

{

printf("%d\n",queue1[i]);

}

}

int main()

{

int ch;

printf("\n 1.Push \n 2.Pop \n 3.Display \n 4.Exit");

while(1)

{

printf("\n Enter your choice"); scanf("%d",&ch);

switch(ch)

{

case 1:

push();

break;

case 2:

pop();

break;

case 3:

display();

break;

case 4:

exit(0);

}

}

return 0;

}

**Output**

# 

# Conclusion: