**Date**: 03/01/2023

**Program 7(b):** Implementation of a simple queue using an array.

**Algorithm:**

**Enqueue**

Step 1: If n-1==rear then this means the queue is already full.

Step 2: But if rear<n means that we can store an element in an array.

Step 3:So increment the rear value by 1 and then insert an element at the rear index.

**Dequeue**

Step 4: If front==-1 or front>rear then no element is available to delete.

Step 5: else delete front index element.

Step 6: if rear==front then set-1 to both front and rear.

Step 7: else we increment front.

**Front**

Step 8: first of all we need to check that queue is not empty.

Step 9: if the queue is empty then we display that the queue is empty we simply return from the function and not execute further inside the function.

Step 10: otherwise, we will return the front index value.

**Display**

Step 11: firstly check whether the queue is not empty.

Step 12:if empty we display that the queue is empty we simply return from the function and not execute further inside the function.

Step 13: else print all elements from front to rear.

**Program code:**

#include<stdio.h>

#include<stdlib.h>

#define maxsize 5

void insert();

void delete();

void display();

int front= -1,rear= -1;

int queue[maxsize];

void insert()

{

int item;

printf("Enter the element:");

scanf("%d",&item);

if(rear==maxsize-1)

{

printf("Overflow\n");

return;

}

if(front==-1&&rear==-1)

{

front= 0;

rear= 0;

}

else

{

rear= rear+1;

}

queue[rear]= item;

printf("Value inserted\n");

}

void delete()

{

int item;

if(front==-1||front>rear)

{

printf("Underflow\n");

return;

}

else

{

item=queue[front];

if(front==rear)

{

front= -1;

rear= -1;

}

else

{

front= front+1;

}

printf("Value deleted\n");

}

}

void display()

{

int i;

if(rear==-1)

{

printf("Empty queue\n");

}

else

{

printf("Printing value\n");

printf("\n");

for(i=front;i<=rear;i++)

{

printf("%d",queue[i]);

}

}

}

int main()

{

int choice;

while(1)

{

printf("\nOperation performed by stack\n");

printf("1. Insert\n");

printf("2. Delete\n");

printf("3. Display\n");

printf("4. Exit\n");

scanf("%d",&choice);

switch(choice)

{

case 1: insert();

break;

case 2: delete();

break;

case 3: display();

break;

case 4: exit(0);

break;

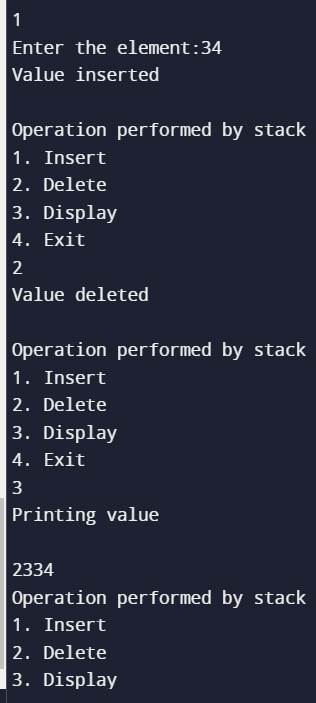
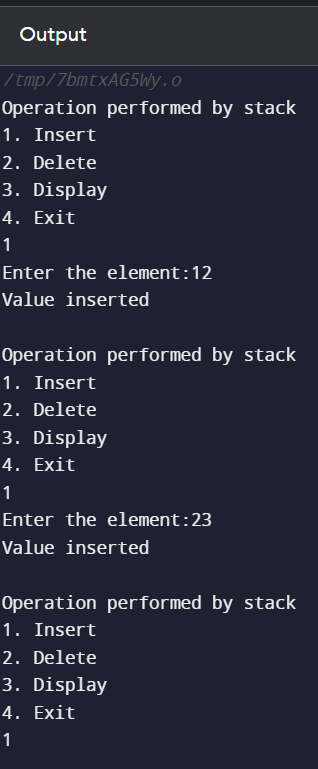
default:printf("Invalid choice\n");

}

}

}

**Output:**



**Link for GitHub:**