```
#include <stdio.h>

#include<stdbool.h>
#include<stdlib.h>
#define MAX 10

int front = -1;
int rear= -1;
int q[MAX];

bool isEmpty()
{
   if(front == -1 && rear == -1) return true;

   else
   return false;
}

bool isFull()
{
   if(rear==MAX-1)
   return true;

   else
   return false;
}

void push(int x)
{
   if(isFull())
   {
      printf("OVERFLOW!");
   }

   else if(isEmpty())
   {
      front = 0;
      rear=0;
      q[rear]=x;
   }

   else
   {
      rear++;
      q[rear]= x;
   }
```

```
}

int pop()
{
   if(isEmpty())
   {
      printf("UNDERFLOW!");
   }

   else
   {
      int val= q[front];
      if(front == rear)
      {
         front = -1;
         rear= -1;
      }
      else
      {
         front++;
      }

      return val;
   }
}

int frontele()
{
   if(isEmpty())
   return -1;

   else
   return q[front];
}



int main() {
   // Write C code here

   printf("This is a menu driven program!\n\n");
   int choice;

   while(1)
   {
      printf("\n\nEnter a choice from 1 to 6:\n1:Push an element\n2:Pop an
element\n3.See the element at the front\n4.Display size of queue\n5.Display the
```

```
queue\n6.Exit the program\n");
        scanf("%d", &choice);

        switch(choice)
        {
            case 1:
            {
                int element;
                printf("Enter an element to be pushed: ");
                scanf("%d", &element);
                push(element);

                printf("The queue after adding %d is: ", element);
                for(int i=front; i<=rear;i++)
                {
                    printf("%d ", q[i]);
                }
                break;
            }

            case 2:
            {
                int val= pop();
                printf("The element popped is: %d", val);

                printf("\n\nThe queue after popping is: ");
                for(int i=front; i<=rear;i++)
                {
                    printf("%d ", q[i]);
                }
                break;
            }

            case 3:
            {
                int f=frontele();
                printf("\n\nThe element at the front of the queue is: %d", f);
                break;
            }

            case 4:
            {
                int size= rear-front+1;
                printf("The size of the queue is: %d", size);
                break;
            }
```

```
            case 5:
            {
               printf("The queue is: ");
               for(int i=front; i<=rear; i++)
               {
                  printf("%d ", q[i]);
               }

               break;
            }

            case 6:
            {
               exit(0);
            }

            default: printf("Invalid input, try again!");
         }

      }

}
```