

```

/* NAME      :MOHAMMED SINAN.P
ROLL.NO     :39
DATE        :24/11/22
PROGRAM     :IMPLEMENTATION OF PRIORITY QUEUE USING ARRAY
INSTITUTION :MES COLLEGE OF ENGINEERING */

```

```

#include<stdio.h>
#include<stdlib.h>
int max,i;
void Display();
void Enqueue();
void Dequeue();

struct pqueue
{
    int data, prio;
} p[10],temp;
int front=-1,rear=-1;
void Enqueue()
{
    if(front == 0 && rear==max-1)
        printf("\nQueue is Full!!\n");
    else if(rear==max-1)
    {
        front=rear=0;
        printf("\nEnter the element and priority : \n");
        scanf("%d %d", &p[rear].data, &p[rear].prio);
    }
    else if(front > 0 && rear == max -1){
        for(i=front;i<=rear;++i){
            p[i-1].data = p[i].data;
            p[i-1].prio = p[i].prio;
        }
        front--;
        rear--;
        rear++;
        printf("\nEnter the element and priority : \n");
        scanf("%d %d", &p[rear].data, &p[rear].prio);
        for(int i=rear;i>front;i--)
        {
            if(p[i].prio > p[i-1].prio)
            {
                temp=p[i-1];
                p[i-1]=p[i];
                p[i]=temp;
            }
        }
    }
    else
    {
        rear++;
        printf("\nEnter the element and priority : \n");
        scanf("%d %d", &p[rear].data, &p[rear].prio);

        for(int i=rear;i>front;i--)
        {

```

```

        if(p[i].prio > p[i-1].prio)
        {
            temp=p[i-1];
            p[i-1]=p[i];
            p[i]=temp;
        }
    }
}

void Display()
{
    if(rear==-1)
    {
        printf("\nQueue is empty\n");
        return;
    }
    else{
        printf("\nQueue elements : ");
        for(int i=front; i<=rear; i++){
            printf("| %d ", p[i].data);
        }
        printf("\nPriority      : ");
        for(int i=front; i<=rear; i++){
            printf("| %d ", p[i].prio);
        }
    }
}

void Dequeue()
{
    if(front==-1)
        printf("\n!!!Queue underflow\n");
    else if(front==rear)
    {
        printf("\nDeleted element is %d of priority %d\n", p[front].data, p[front].prio);
        front=rear=-1;
    }
    else
    {
        printf("\nDeleted element is %d of priority %d\n", p[front].data, p[front].prio);
        front++;
    }
}

void main()
{
    int choice;

    printf("\n-PRIORITY QUEUE-\n");
    printf("\nEnter the size of queue : ");
    scanf("%d", &max);
    printf("\n1.Enqueue\n2.Dequeue\n3.Display\n4.Exit\n");
    do{
        printf("\nEnter your choice : ");
        scanf("%d", &choice);
        switch(choice)
        {
            case 1: Enqueue();
                    break;

```

```

        case 2: Dequeue();
                break;
        case 3: Display();
                break;
        case 4: printf("\n-EXITED FROM MENU-");
                break;
        default: printf("Invalid choice\n");
    }
}while(choice != 4);
}

```

Output :--

Enter the size of queue : 2

1.Enqueue
2.Dequeue
3.Display
4.Exit

Enter your choice : 1

Enter the element and priority :

3

5

Enter your choice : 1

Enter the element and priority :

4

8

Enter your choice : 1

Queue is Full!!

Enter your choice : 3

Queue elements : | 4 | 3

Priority : | 8 | 5

Enter your choice : 2

Deleted element is 4 of priority 8

Enter your choice : 2

Deleted element is 3 of priority 5

Enter your choice : 2

!!!Queue underflow

Enter your choice : 4

-EXITED FROM MENU-