

```
1: #include <stdio.h>
2: #include<stdlib.h>
3: struct node{
4:     int data;
5:     struct node *next;
6: };
7: struct node *front,*rare;
8: void enqueue(int x)
9: {
10:     struct node *newnode;
11:     newnode=(struct node*)malloc(sizeof(struct node));
12:     newnode->next=0;
13:     newnode->data=x;
14:     if(front==0 && rare==0)
15:     {
16:         front=rare=newnode;
17:     }
18:     else
19:     {
20:         rare->next=newnode;
21:         rare=newnode;
22:     }
23: }
24: void dequeue()
25: {
26:     struct node *temp;
27:     temp=front;
28:     if(front==0 && rare==0)
29:     {
30:         printf("Queue is empty \n");
31:     }
32:     else
33:     {
34:         front=front->next;
35:         free(temp);
36:     }
37: }
38: }
39: void display()
```

```
40: {
41:     struct node *temp;
42:     temp=front;
43:     if(front==0 &&rare==0)
44:     {
45:         printf("Queue is empty \n");
46:     }
47:     else
48:     {
49:         while(temp!=0)
50:         {
51:             printf("%d \t",temp->data);
52:             temp=temp->next;
53:         }
54:     }
55:     printf("\n");
56: }
57: void main()
58: {
59:     enqueue(1);
60:     enqueue(2);
61:     enqueue(3);
62:     enqueue(4);
63:     enqueue(5);
64:     display();
65:     dequeue();
66:     display();
67: }
68:
```