

```

1  #include <stdio.h>
2  #include <stdlib.h>
3  struct node
4  {
5      int data;
6      struct node *next;
7  };
8  void qinsert();
9  void qdisplay();
10 void qdel();
11 void spush();
12 void spop();
13 void sdisplay();
14 struct node *rear=NULL, *front =NULL, *top=NULL;
15
16 int main(int argc, char **argv)
17 {
18
19     int choice;
20     printf("Enter the choice\n1.Stack\n2.Queue\n");
21     scanf("%d",&choice);
22
23     if(choice==1)
24     do
25     { printf("\n1. Push \n2. Display \n3. Pop\n");
26       printf("\nEnter your choice : ");
27       scanf("%d",&choice);
28       switch(choice)
29       {
30           case 1: spush(); break;
31           case 2: sdisplay();break;
32           case 3: spop(); break;
33           default: if(choice!=4)
34                   printf("\nInvalid Input");
35       }
36     }while(choice!=4);
37
38     else if(choice==2)
39     do
40     { printf("\nQueue implementation using linked list\n");
41       printf("\n1. Create \n2. Display \n3. Delete \n4. Exit\n");
42       printf("\nEnter your choice : ");
43       scanf("%d",&choice);
44       switch(choice)
45       { case 1: qinsert(); break;
46         case 2: qdisplay();break;
47         case 3: qdel(); break;
48         default: if(choice!=4)
49                 printf("\nInvalid Input");
50       }
51     }while(choice!=4);
52 }
53
54 void qinsert()
55 {
56     struct node *newnode;
57     newnode=(struct node *) malloc(sizeof(struct node));
58     printf("Enter the element:\n");
59     scanf("%d",&newnode->data);
60     newnode->next=NULL;
61
62     if(rear==NULL)
63     {
64         rear=newnode;
65         front=newnode;
66     }
67 }
68

```

```

clang-7 -pthread -lm -o main main.c
./main
Enter the choice
1.Stack
2.Queue
1

1. Push
2. Display
3. Pop

Enter your choice : 1
Enter the element
1

1. Push
2. Display
3. Pop

Enter your choice : 1
Enter the element
2

1. Push
2. Display
3. Pop

Enter your choice : 1
Enter the element
3

1. Push
2. Display

```

```

35     }
36
37     }while(choice!=4);
38
39     else if(choice==2)
40     do
41     { printf("\nQueue implementation using linked list\n");
42       printf("\n1. Create \n2. Display \n3. Delete \n4. Exit\n");
43       printf("\nEnter your choice : ");
44       scanf("%d",&choice);
45       switch(choice)
46       { case 1: qinsert(); break;
47         case 2: qdisplay();break;
48         case 3: qdel(); break;
49         default: if(choice!=4)
50                 printf("\nInvalid Input");
51       }
52     }while(choice!=4);
53 }
54
55 void qinsert()
56 {
57     struct node *newnode;
58     newnode=(struct node *) malloc(sizeof(struct node));
59     printf("Enter the element:\n");
60     scanf("%d",&newnode->data);
61     newnode->next=NULL;
62
63     if(rear==NULL)
64     {
65         rear=newnode;
66         front=newnode;
67     }
68 }
69

```

```

1. Push
2. Display
3. Pop

Enter your choice : 2
321

1. Push
2. Display
3. Pop

Enter your choice : 3
element removed is 3:
1. Push
2. Display
3. Pop

Enter your choice : 3
element removed is 2:
1. Push
2. Display
3. Pop

Enter your choice : 3
element removed is 1:
1. Push
2. Display
3. Pop

Enter your choice : 3
stack is empty
1. Push
2. Display

```

```

68     }
69     else
70     {
71         rear->next=newnode;
72         rear=newnode;
73     }
74 }
75
76 void qdel()
77 {
78     if(front==NULL)
79     {
80         printf("Queue is empty\n");return;
81     }
82     else
83     {
84         printf("Deleted ele is %d",front->data);
85         if(front==rear)
86         {
87             printf("Queue is empty\n");
88             front=NULL; rear=NULL;
89         }
90         else
91         {
92             front=front->next;
93         }
94     }
95 }
96
97 void qdisplay()
98 {
99     struct node *temp;
100     if(front ==NULL)
101     {
102         printf("Queue is empty");
103     }
104 }
105

```

```

clang-7 -pthread -lm -o main main.c
./main
Enter the choice
1.Stack
2.Queue
2

Queue implementation using linked list

1. Create
2. Display
3. Delete
4. Exit

Enter your choice : 1
Enter the element:
1

Queue implementation using linked list

1. Create
2. Display
3. Delete
4. Exit

Enter your choice : 1
Enter the element:
2

Queue implementation using linked list

1. Create
2. Display

```

```

101     printf("Queue is empty");
102     return;
103 }
104 temp=front;
105 while (temp !=NULL)
106 {
107     printf("%d ",temp->data);
108     temp=temp->next;
109 }
110 }
111 }
112 void spush()
113 {
114     int item;
115     struct node *newnode;
116     printf("Enter the element\n");
117     scanf("%d",&item);
118
119     newnode=(struct node*)malloc(sizeof(struct node));
120     newnode->data=item;
121     newnode->next=NULL;
122     if(top==NULL)
123         top=newnode;
124     else
125         newnode->next=top;
126         top=newnode;
127 }
128 void spop()
129 {
130     if(top==NULL)
131         printf("stack is empty");
132     else
133     {
134

```

Queue implementation using linked list

1. Create
2. Display
3. Delete
4. Exit

Enter your choice : 1
Enter the element:
3

Queue implementation using linked list

1. Create
2. Display
3. Delete
4. Exit

Enter your choice : 2
1 2 3

Queue implementation using linked list

1. Create
2. Display
3. Delete
4. Exit

Enter your choice : 3
Deleted ele is 1

Queue implementation using linked list

1. Create
2. Display

```

27 }
28 void spop()
29 {
30     if(top==NULL)
31         printf("stack is empty");
32     else
33     {
34
35         printf("element removed is %d:", top->data);
36
37         top=top->next;
38     }
39 }
40
41 }
42 }
43
44 void sdisplay()
45 {
46     struct node *temp;
47     temp=top;
48     if(top==NULL)
49         printf("Stack is empty");
50     while(temp!=NULL)
51     {
52         printf("%d",temp->data);
53         temp=temp->next;
54     }
55 }
56 }

```

3. Delete
4. Exit

Enter your choice : 2
1 2 3

Queue implementation using linked list

1. Create
2. Display
3. Delete
4. Exit

Enter your choice : 3
Deleted ele is 1

Queue implementation using linked list

1. Create
2. Display
3. Delete
4. Exit

Enter your choice : 3
Deleted ele is 2

Queue implementation using linked list

1. Create
2. Display
3. Delete
4. Exit

Enter your choice : 3
Deleted ele is 3

Queue is empty