```
        else
        {
            temp->prev->next=temp->next;
            temp->next->prev=temp->prev;
        }
}
void display()
{
        struct node *ptr;
        ptr=head;
        while(ptr!=NULL)
        {
            printf("%d\t",ptr->data);
            ptr=ptr->next;
        }
        printf("\n");
}


int main()
{
    int choice;

        while(1)
        {
                printf(" 1. Insert at the left \n");
                printf(" 2. Insert at the right \n");
                printf(" 3. Delete \n");
```

```
clang-7 -pthread -lm -o main main.c
./main
 1. Insert at the left
 2. Insert at the right
 3. Delete
 4. Display
 5. Exit
Enter your choice
1
Enter the item
4
 1. Insert at the left
 2. Insert at the right
 3. Delete
 4. Display
 5. Exit
Enter your choice
1
Enter the item
3
 1. Insert at the left
 2. Insert at the right
 3. Delete
 4. Display
 5. Exit
Enter your choice
1
Enter the item
2
 1. Insert at the left
```


```
        else
        {
            temp->prev->next=temp->next;
            temp->next->prev=temp->prev;
        }
}
void display()
{
        struct node *ptr;
        ptr=head;
        while(ptr!=NULL)
        {
            printf("%d\t",ptr->data);
            ptr=ptr->next;
        }
        printf("\n");
}


int main()
{
    int choice;

        while(1)
        {
                printf(" 1. Insert at the left \n");
                printf(" 2. Insert at the right \n");
                printf(" 3. Delete \n");
```


```
Enter the item
2
 1. Insert at the left
 2. Insert at the right
 3. Delete
 4. Display
 5. Exit
Enter your choice
1
Enter the item
1
 1. Insert at the left
 2. Insert at the right
 3. Delete
 4. Display
 5. Exit
Enter your choice
4
1   2   3   4
 1. Insert at the left
 2. Insert at the right
 3. Delete
 4. Display
 5. Exit
Enter your choice
3
Enter the element to be deleted
4
 1. Insert at the left
 2. Insert at the right
```

```
88        else
89        {
90            temp->prev->next=temp->next;
91            temp->next->prev=temp->prev;
92        }
93    }
94    void display()
95    {
96        struct node *ptr;
97        ptr=head;
98        while(ptr!=NULL)
99        {
100           printf("%d\t",ptr->data);
101           ptr=ptr->next;
102       }
103       printf("\n");
104   }
105
106
107   int main()
108   {
109       int choice;
110
111       while(1)
112       {
113               printf(" 1. Insert at the left \n");
114               printf(" 2. Insert at the right \n");
115               printf(" 3. Delete \n");
```

```
 1. Insert at the left
 2. Insert at the right
 3. Delete
 4. Display
 5. Exit
Enter your choice
4
1   2   3
 1. Insert at the left
 2. Insert at the right
 3. Delete
 4. Display
 5. Exit
Enter your choice
3
Enter the element to be deleted
3
 1. Insert at the left
 2. Insert at the right
 3. Delete
 4. Display
 5. Exit
Enter your choice
4
1   2
 1. Insert at the left
 2. Insert at the right
 3. Delete
 4. Display
 5. Exit
```

```
88        else
89        {
90            temp->prev->next=temp->next;
91            temp->next->prev=temp->prev;
92        }
93    }
94    void display()
95    {
96        struct node *ptr;
97        ptr=head;
98        while(ptr!=NULL)
99        {
100           printf("%d\t",ptr->data);
101           ptr=ptr->next;
102       }
103       printf("\n");
104   }
105
106
107   int main()
108   {
109       int choice;
110
111       while(1)
112       {
113               printf(" 1. Insert at the left \n");
114               printf(" 2. Insert at the right \n");
115               printf(" 3. Delete \n");
```

```
Enter your choice
3
Enter the element to be deleted
2
 1. Insert at the left
 2. Insert at the right
 3. Delete
 4. Display
 5. Exit
Enter your choice
4
1
 1. Insert at the left
 2. Insert at the right
 3. Delete
 4. Display
 5. Exit
Enter your choice
3
Enter the element to be deleted
1
 1. Insert at the left
 2. Insert at the right
 3. Delete
 4. Display
 5. Exit
Enter your choice
4

 1. Insert at the left
```