Lab program_6

WAP to Implement Singly Linked List with following operations

a) Create a linked list.

b) Insertion of a node at first position, at any position and at end of list.

c) Deletion of first element, specified element and last element in the list.

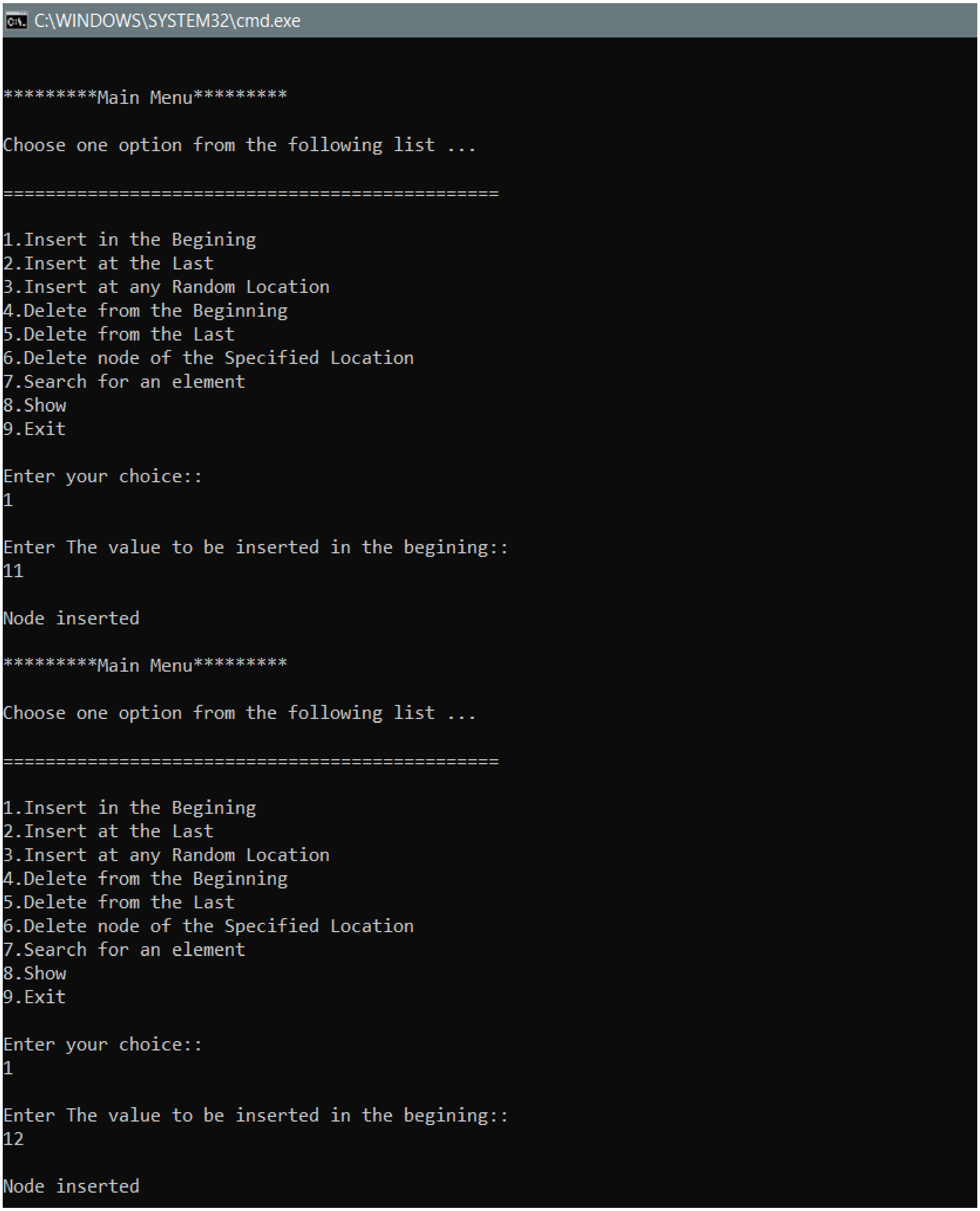
d) Display the contents of the linked list.

Solution:

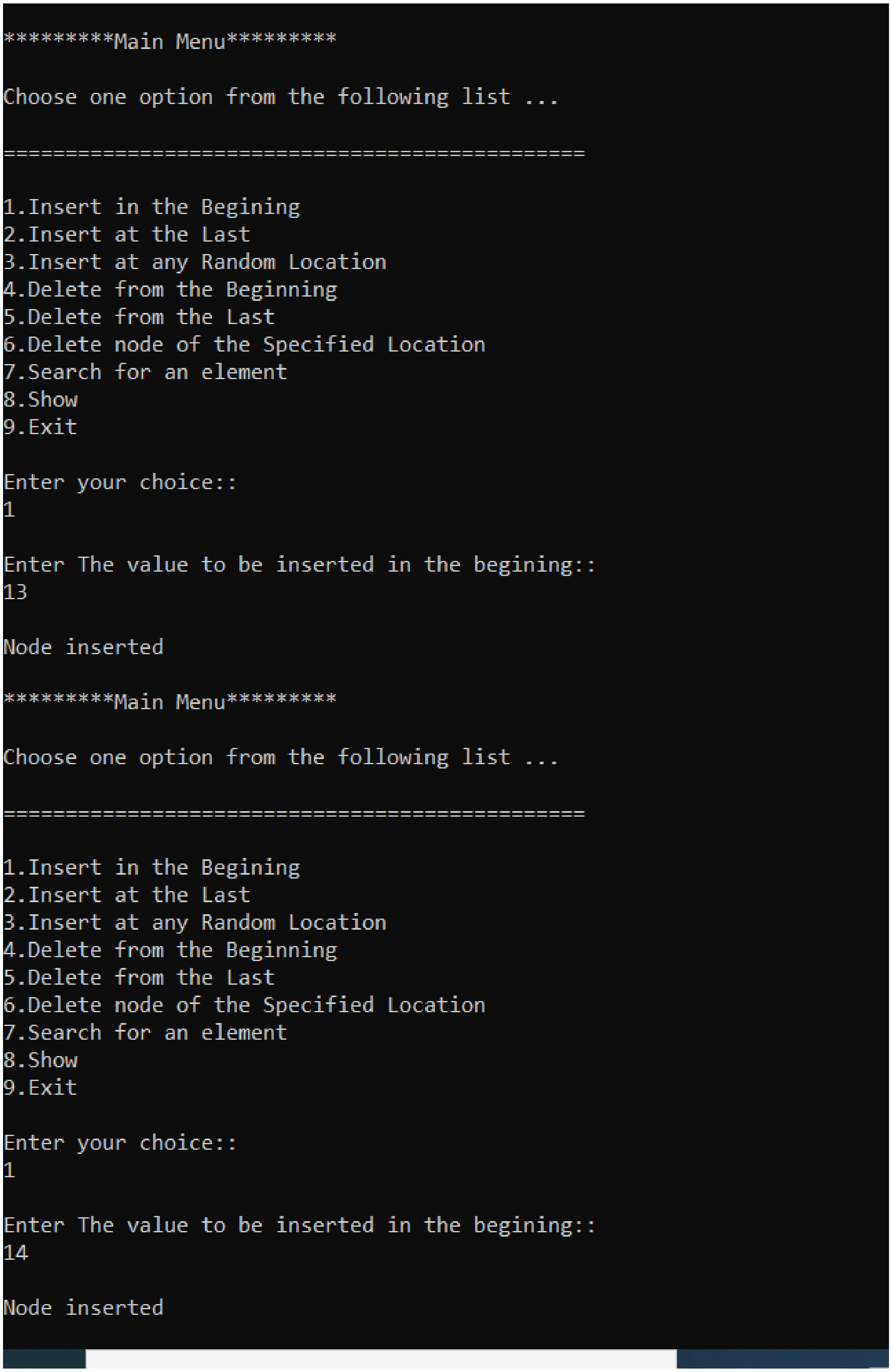
```
#include<stdio.h>
#include<stdlib.h>
struct node
{
   int data;
   struct node *next;
};
struct node *head;

void beginsert ();
void lastinsert ();
void randominsert();
void begin_delete();
void last_delete();
void random_delete();
void display();
void search();
int main ()
{
 int choice =0;

 while(choice != 9)

 {

    printf("\n\n*********Main  Menu*********\n");

    printf("\nChoose one option from the following list ...\n");

    printf("\n===============================================\n");

    printf("\n1.Insert in the Begining\n2.Insert at the Last\n3.Insert at any Random Location\n4.Delete from the Beginning\n5.Delete from the Last\n6.Delete node of the Specified Location\n7.Search for an element\n8.Show\n9.Exit\n");

    printf("\nEnter your choice::\n");

    scanf("\n%d",&choice);

    switch(choice)

    {

       case 1:

       beginsert();

       break;

       case 2:

       lastinsert();

       break;

       case 3:

       randominsert();

       break;

       case 4:

       begin_delete();

       break;

       case 5:

       last_delete();

       break;

       case 6:
            random_delete();
            break;
            case 7:
            search();
            break;
            case 8:
            display();
            break;
            case 9:
            exit(0);
            break;
            default:
            printf("Please enter valid choice..");
        }
    }
}
void beginsert()
{
    struct node *ptr;
    int item;
    ptr = (struct node *) malloc(sizeof(struct node *));
    if(ptr == NULL)
    {
        printf("\nOVERFLOW");
    }
    else
    {
        printf("\nEnter  The value to be inserted in the begining::\n");
        scanf("%d",&item);
        ptr->data = item;
        ptr->next = head;
        head = ptr;
        printf("\nNode inserted");
    }

}
void lastinsert()
{
    struct node *ptr,*temp;
    int item;
    ptr = (struct node*)malloc(sizeof(struct node));
    if(ptr == NULL)
    {
        printf("\nOVERFLOW");
    }
    else
    {
        printf("\nEnter The value to be inserted in the last::\n");
        scanf("%d",&item);
        ptr->data = item;
        if(head == NULL)
        {
            ptr -> next = NULL;
            head = ptr;
            printf("\nNode inserted");
        }
        else
        {
            temp = head;
            while (temp -> next != NULL)
            {
                temp = temp -> next;
            }
            temp->next = ptr;
            ptr->next = NULL;
            printf("\nNode inserted");

        }
    }
}
void randominsert()
{
    int i,loc,item;
    struct node *ptr, *temp;
    ptr = (struct node *) malloc (sizeof(struct node));
    if(ptr == NULL)
    {
        printf("\nOVERFLOW");
    }
    else
    {
        printf("\nEnter the element value to be inserted at Random Location::");
        scanf("%d",&item);
        ptr->data = item;
        printf("\nEnter the location after which you want to insert:: ");
        scanf("\n%d",&loc);
        temp=head;
        for(i=0;i<loc;i++)
        {
            temp = temp->next;
            if(temp == NULL)
            {
                printf("\ncan't insert\n");
                return;
            }

        }
        ptr ->next = temp ->next;
        temp ->next = ptr;
        printf("\nNode inserted");
    }
}
void begin_delete()
{
    struct node *ptr;
    if(head == NULL)
    {
        printf("\nList is empty\n");
    }
    else
    {
        ptr = head;
        head = ptr->next;
        free(ptr);
        printf("\nNode deleted from the begining ...\n");
    }
```

```
}
void last_delete()
{
    struct node *ptr,*ptr1;
    if(head == NULL)
    {
        printf("\nlist  is empty");
    }
    else if(head -> next == NULL)
    {
        head = NULL;
        free(head);
        printf("\nOnly  node of the list deleted ...\n");
    }

    else
    {
        ptr = head;
        while(ptr->next  != NULL)
        {
            ptr1 = ptr;
            ptr = ptr ->next;
        }
        ptr1->next = NULL;
        free(ptr);
        printf("\nDeleted  Node from the last ...\n");
    }
}
void random_delete()
```

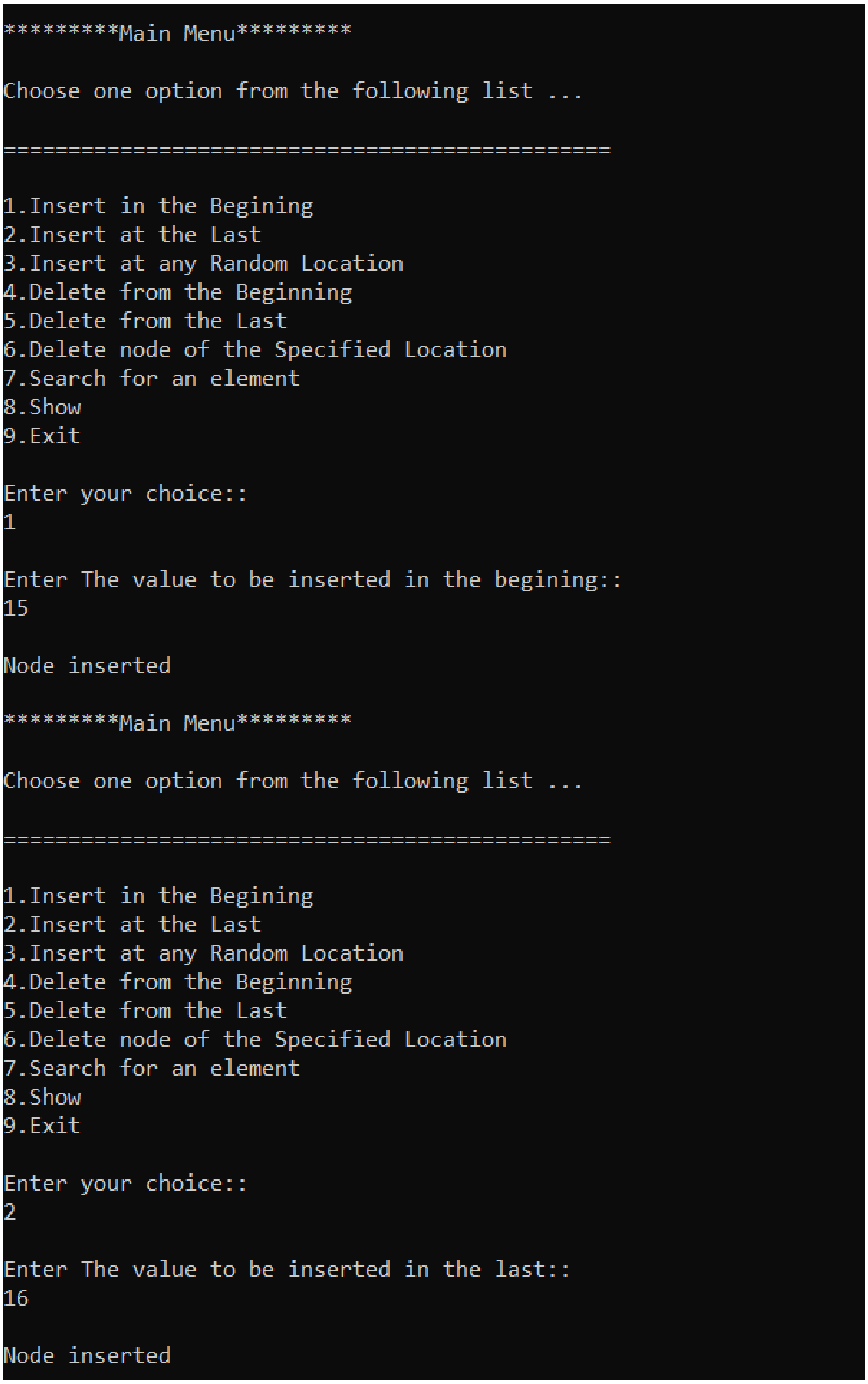
```
{
    struct node *ptr,*ptr1;
    int loc,i;
    printf("\n  Enter the location of the node after which you want to perform deletion \n");
    scanf("%d",&loc);
    ptr=head;
    for(i=0;i<loc;i++)
    {
        ptr1 = ptr;
        ptr = ptr->next;

        if(ptr == NULL)
        {
            printf("\nCan't  delete");
            return;
        }
    }
    ptr1 ->next = ptr ->next;
    free(ptr);
    printf("\nDeleted  node %d ",loc+1);
}
void search()
{
    struct node *ptr;
    int item,i=0,flag;
    ptr = head;
    if(ptr == NULL)
    {
        printf("\nEmpty  List\n");
    }
    else
    {
        printf("\nEnter  item which you want to search::\n");
        scanf("%d",&item);
        while (ptr!=NULL)
        {
            if(ptr->data == item)
            {
                printf("item  found at location %d ",i+1);
                flag=0;
            }
            else
            {
                flag=1;
            }
            i++;
            ptr = ptr -> next;
        }
        if(flag==1)
        {
            printf("Item  not found\n");
        }
    }

}

void display()
{
    struct node *ptr;
    ptr = head;
    if(ptr == NULL)
    {
        printf("Nothing to print");
    }
    else
    {
        printf("\nprinting values . . . . .\n");
        while (ptr!=NULL)
        {
            printf("\n%d",ptr->data);
            ptr = ptr -> next;
        }
    }
}
```



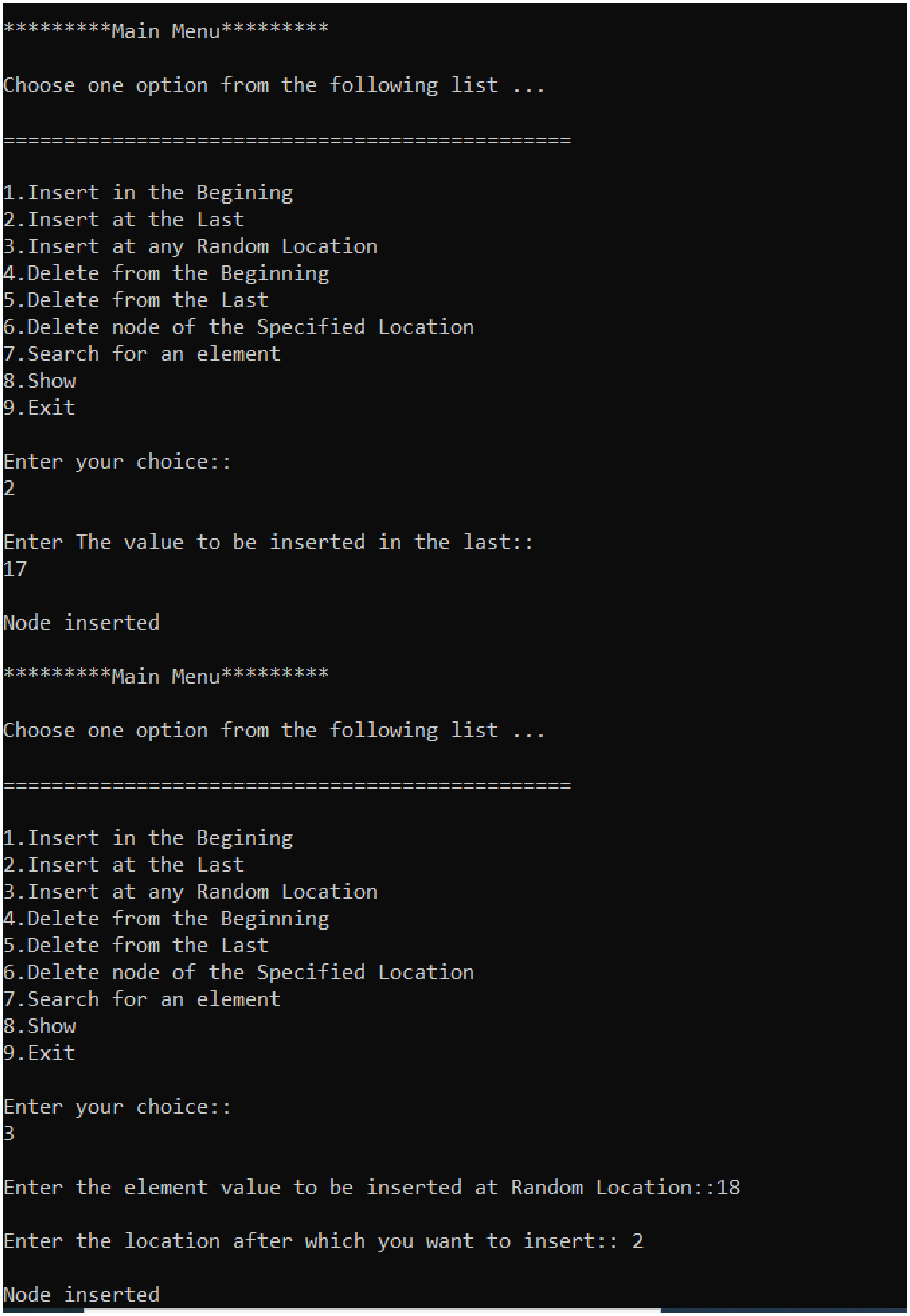
```
*********Main Menu*********

Choose one option from the following list ...

===============================================

1.Insert in the Begining
2.Insert at the Last
3.Insert at any Random Location
4.Delete from the Beginning
5.Delete from the Last
6.Delete node of the Specified Location
7.Search for an element
8.Show
9.Exit

Enter your choice::
1

Enter The value to be inserted in the begining::
11

Node inserted

*********Main Menu*********

Choose one option from the following list ...

===============================================

1.Insert in the Begining
2.Insert at the Last
3.Insert at any Random Location
4.Delete from the Beginning
5.Delete from the Last
6.Delete node of the Specified Location
7.Search for an element
8.Show
9.Exit

Enter your choice::
1

Enter The value to be inserted in the begining::
12

Node inserted
```



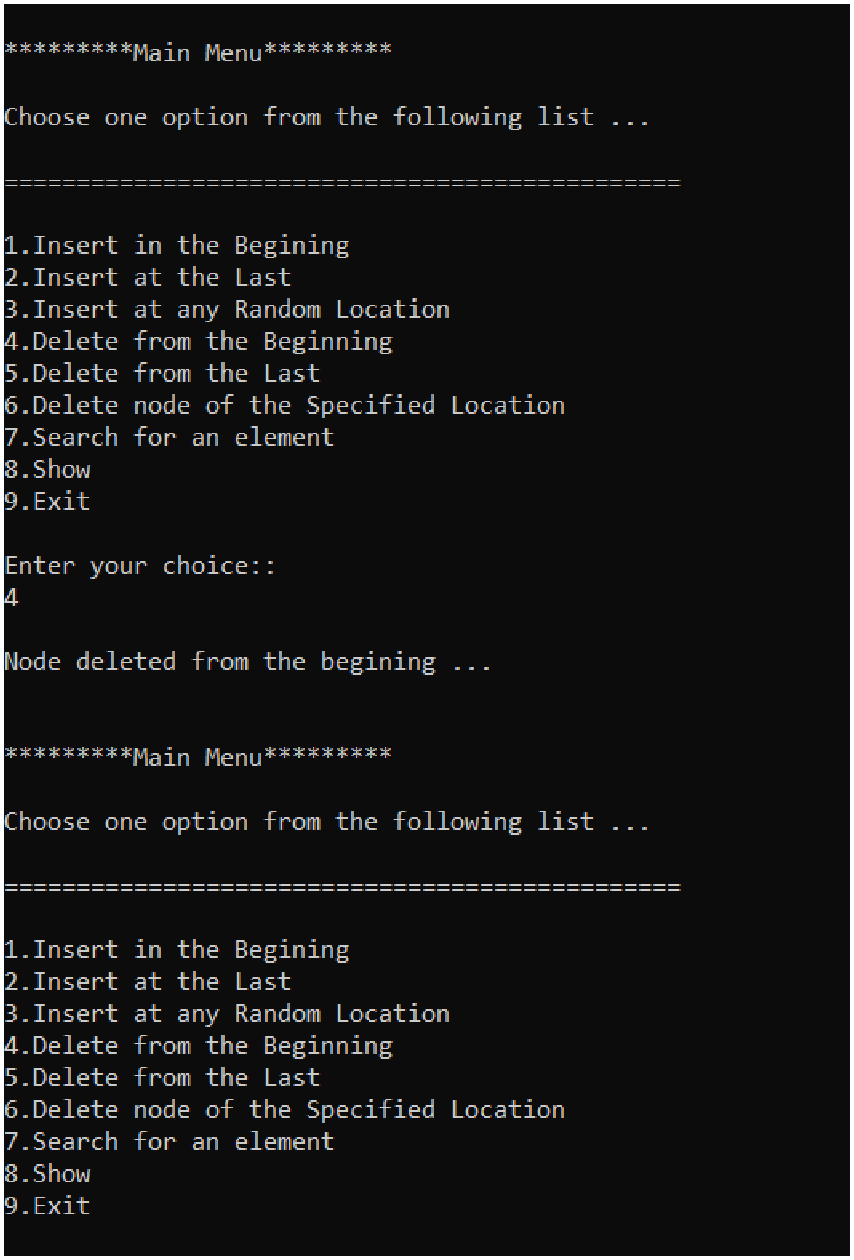
```
*********Main Menu*********

Choose one option from the following list ...

===============================================

1.Insert in the Begining
2.Insert at the Last
3.Insert at any Random Location
4.Delete from the Beginning
5.Delete from the Last
6.Delete node of the Specified Location
7.Search for an element
8.Show
9.Exit

Enter your choice::
1

Enter The value to be inserted in the begining::
13

Node inserted

*********Main Menu*********

Choose one option from the following list ...

===============================================

1.Insert in the Begining
2.Insert at the Last
3.Insert at any Random Location
4.Delete from the Beginning
5.Delete from the Last
6.Delete node of the Specified Location
7.Search for an element
8.Show
9.Exit

Enter your choice::
1

Enter The value to be inserted in the begining::
14

Node inserted
```



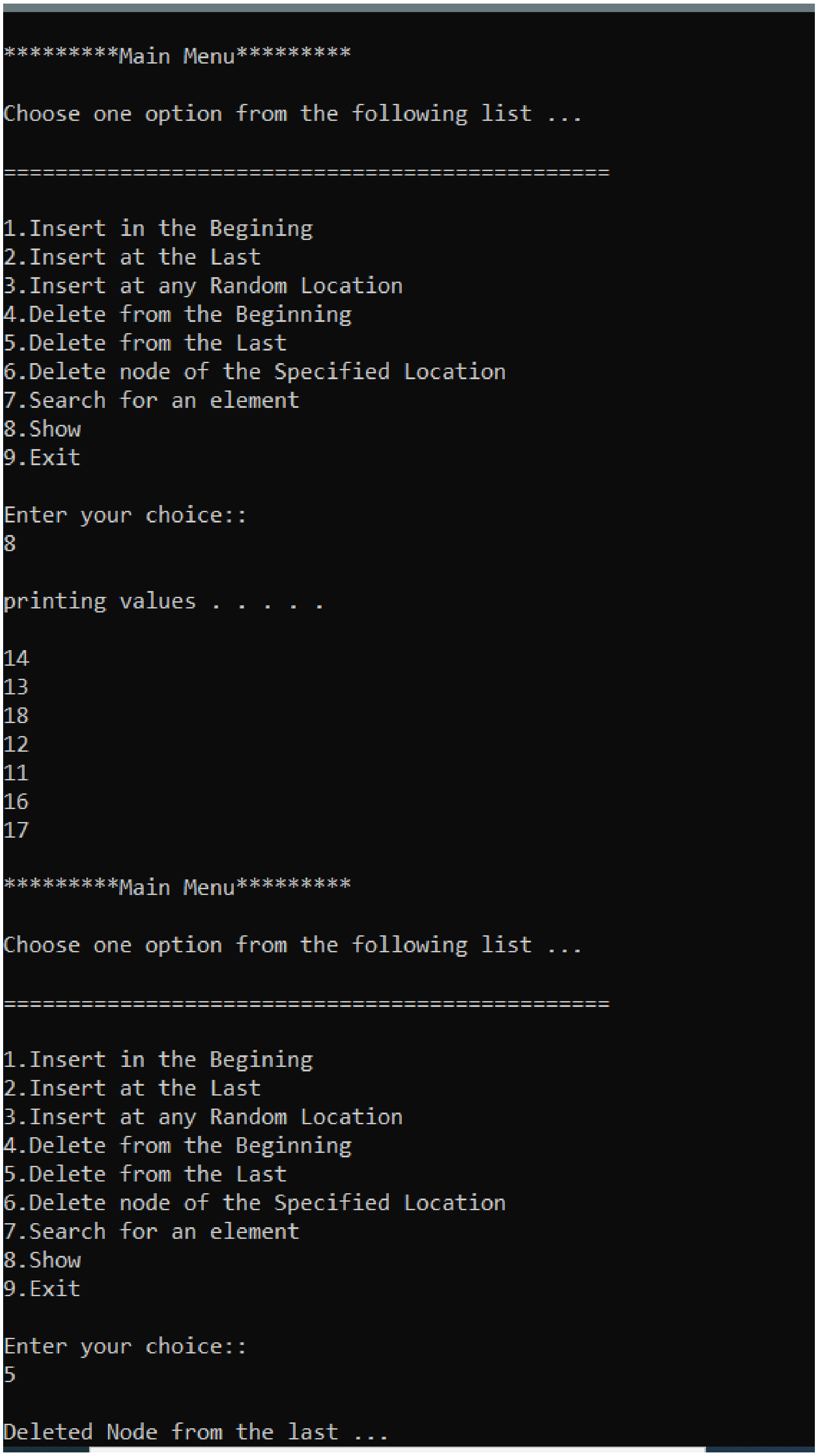
```
*********Main Menu*********

Choose one option from the following list ...

===============================================

1.Insert in the Begining
2.Insert at the Last
3.Insert at any Random Location
4.Delete from the Beginning
5.Delete from the Last
6.Delete node of the Specified Location
7.Search for an element
8.Show
9.Exit

Enter your choice::
1

Enter The value to be inserted in the begining::
15

Node inserted

*********Main Menu*********

Choose one option from the following list ...

===============================================

1.Insert in the Begining
2.Insert at the Last
3.Insert at any Random Location
4.Delete from the Beginning
5.Delete from the Last
6.Delete node of the Specified Location
7.Search for an element
8.Show
9.Exit

Enter your choice::
2

Enter The value to be inserted in the last::
16

Node inserted
```



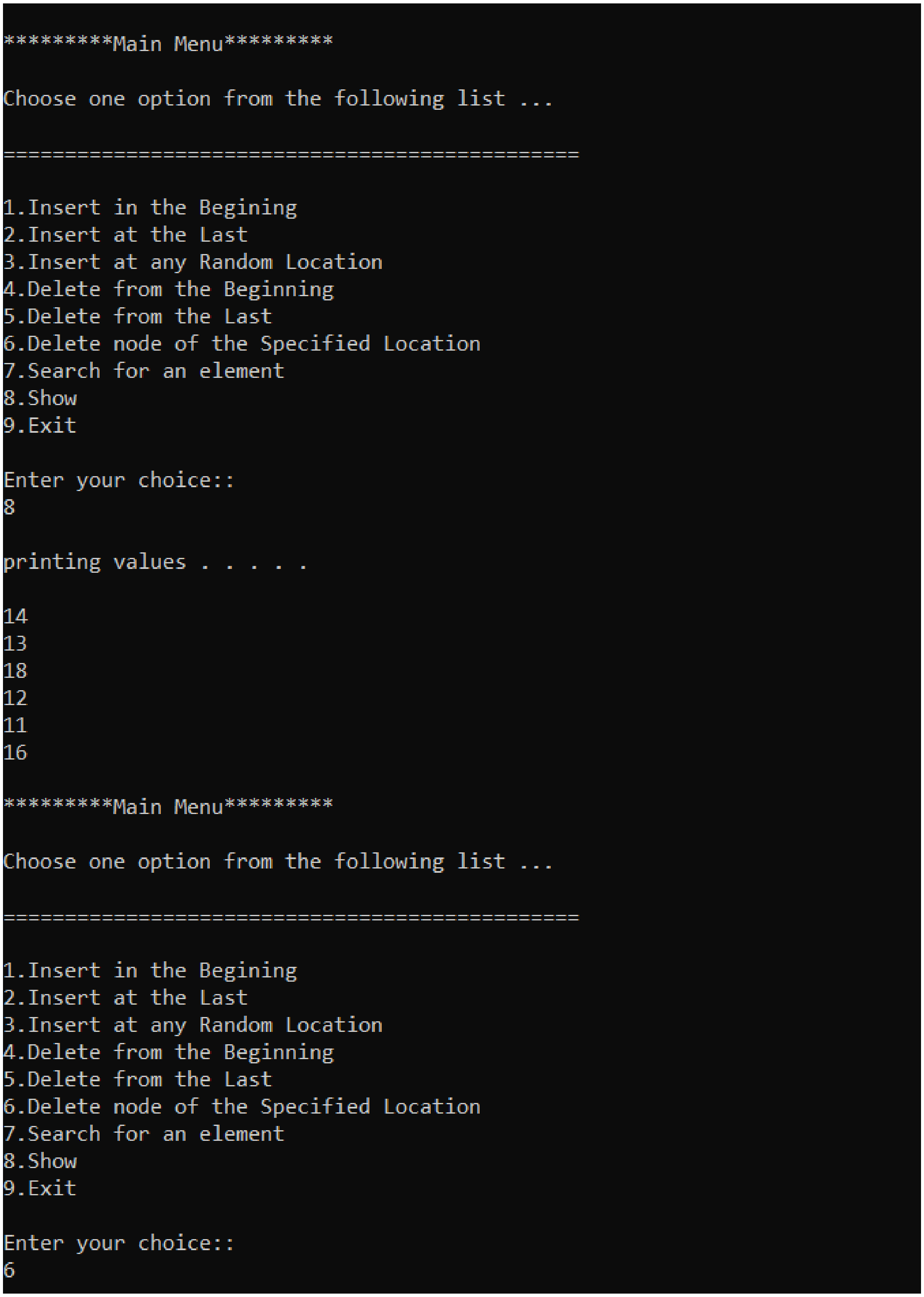
```
*********Main Menu*********

Choose one option from the following list ...

===============================================

1.Insert in the Begining
2.Insert at the Last
3.Insert at any Random Location
4.Delete from the Beginning
5.Delete from the Last
6.Delete node of the Specified Location
7.Search for an element
8.Show
9.Exit

Enter your choice::
2

Enter The value to be inserted in the last::
17

Node inserted

*********Main Menu*********

Choose one option from the following list ...

===============================================

1.Insert in the Begining
2.Insert at the Last
3.Insert at any Random Location
4.Delete from the Beginning
5.Delete from the Last
6.Delete node of the Specified Location
7.Search for an element
8.Show
9.Exit

Enter your choice::
3

Enter the element value to be inserted at Random Location::18

Enter the location after which you want to insert:: 2

Node inserted
```



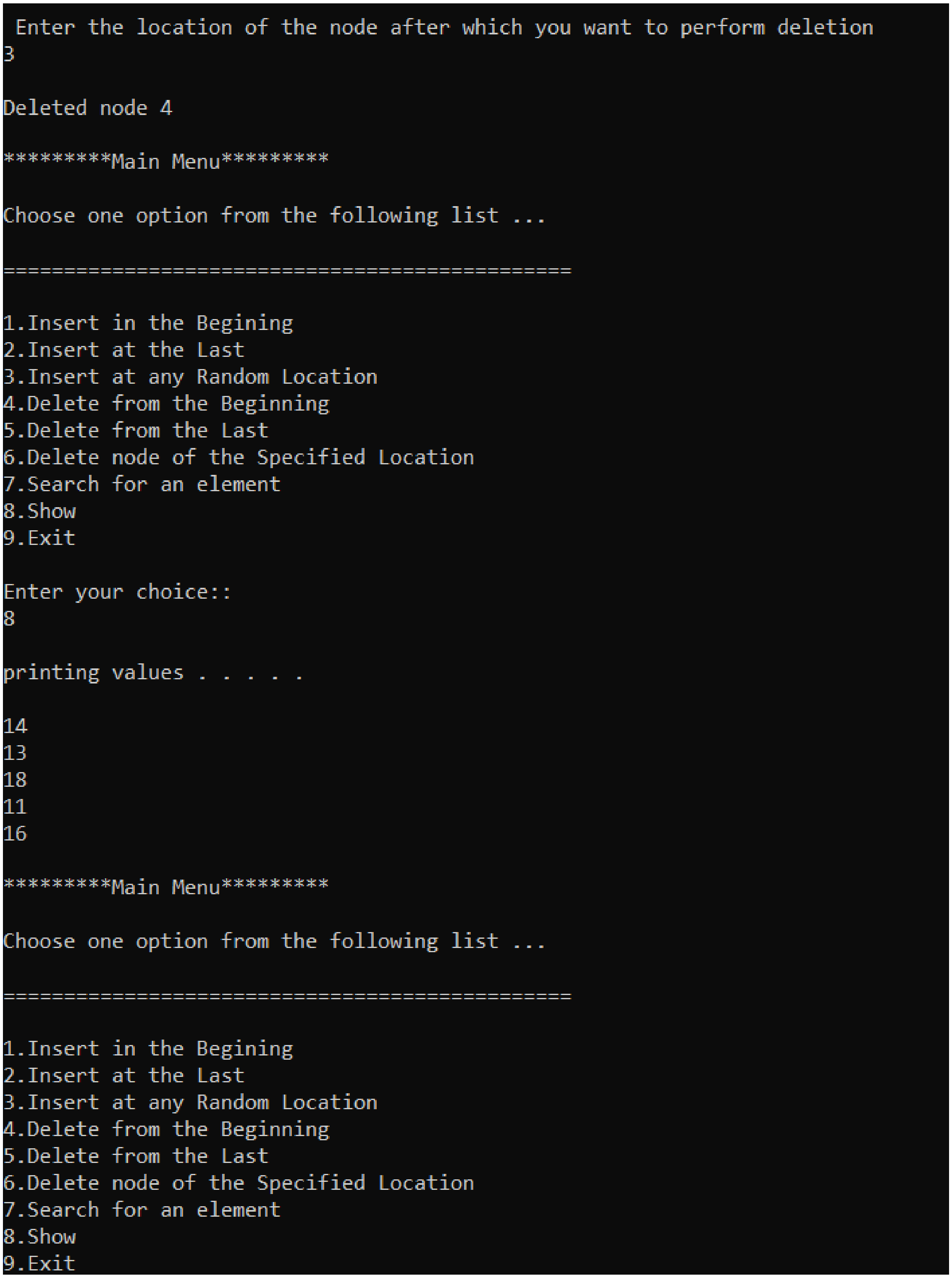
```
*********Main Menu*********

Choose one option from the following list ...

===============================================

1.Insert in the Begining
2.Insert at the Last
3.Insert at any Random Location
4.Delete from the Beginning
5.Delete from the Last
6.Delete node of the Specified Location
7.Search for an element
8.Show
9.Exit

Enter your choice::
4

Node deleted from the begining ...


*********Main Menu*********

Choose one option from the following list ...

===============================================

1.Insert in the Begining
2.Insert at the Last
3.Insert at any Random Location
4.Delete from the Beginning
5.Delete from the Last
6.Delete node of the Specified Location
7.Search for an element
8.Show
9.Exit
```



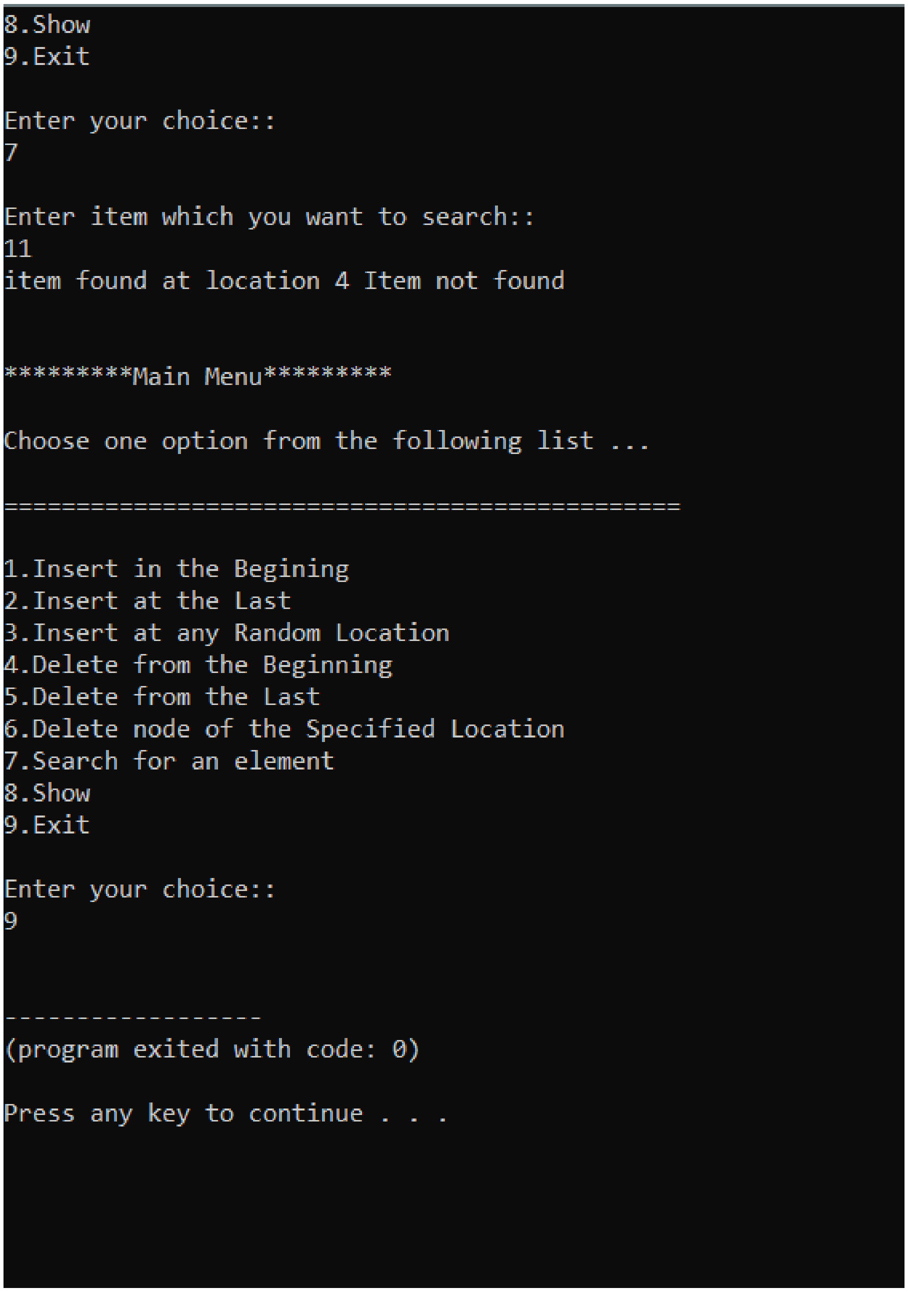
```
*********Main Menu*********

Choose one option from the following list ...

===============================================

1.Insert in the Begining
2.Insert at the Last
3.Insert at any Random Location
4.Delete from the Beginning
5.Delete from the Last
6.Delete node of the Specified Location
7.Search for an element
8.Show
9.Exit

Enter your choice::
8

printing values . . . . .

14
13
18
12
11
16
17

*********Main Menu*********

Choose one option from the following list ...

===============================================

1.Insert in the Begining
2.Insert at the Last
3.Insert at any Random Location
4.Delete from the Beginning
5.Delete from the Last
6.Delete node of the Specified Location
7.Search for an element
8.Show
9.Exit

Enter your choice::
5

Deleted Node from the last ...
```



```
*********Main Menu*********

Choose one option from the following list ...

===============================================

1.Insert in the Begining
2.Insert at the Last
3.Insert at any Random Location
4.Delete from the Beginning
5.Delete from the Last
6.Delete node of the Specified Location
7.Search for an element
8.Show
9.Exit

Enter your choice::
8

printing values . . . . .

14
13
18
12
11
16

*********Main Menu*********

Choose one option from the following list ...

===============================================

1.Insert in the Begining
2.Insert at the Last
3.Insert at any Random Location
4.Delete from the Beginning
5.Delete from the Last
6.Delete node of the Specified Location
7.Search for an element
8.Show
9.Exit

Enter your choice::
6
```



```
 Enter the location of the node after which you want to perform deletion
3

Deleted node 4

*********Main Menu*********

Choose one option from the following list ...

===============================================

1.Insert in the Begining
2.Insert at the Last
3.Insert at any Random Location
4.Delete from the Beginning
5.Delete from the Last
6.Delete node of the Specified Location
7.Search for an element
8.Show
9.Exit

Enter your choice::
8

printing values . . . . .

14
13
18
11
16

*********Main Menu*********

Choose one option from the following list ...

===============================================

1.Insert in the Begining
2.Insert at the Last
3.Insert at any Random Location
4.Delete from the Beginning
5.Delete from the Last
6.Delete node of the Specified Location
7.Search for an element
8.Show
9.Exit
```



```
8.Show
9.Exit

Enter your choice::
7

Enter item which you want to search::
11
item found at location 4 Item not found


*********Main Menu*********

Choose one option from the following list ...

===============================================

1.Insert in the Begining
2.Insert at the Last
3.Insert at any Random Location
4.Delete from the Beginning
5.Delete from the Last
6.Delete node of the Specified Location
7.Search for an element
8.Show
9.Exit

Enter your choice::
9



------------------
(program exited with code: 0)

Press any key to continue . . .
```