Institute of Computer Technology
B. Tech Computer Science and Engineering
Subject: DS (2CSE302)

PRACTICAL-13

AIM: - Implement the scenario based on linked list.

Ques. Inspire.Pvt.Ltd company is selling soft toys for children. Ridham is working in HR department of Inspire.Pvt.Ltd company. She needs to take order from the client however sometimes clients are asked to her for prepare details in ascending order after given order of some items. So as per the requirement of client she needs to the details of order as per following scenario. Refer it and provide appropriate solution for it:

1. Insert new item at the first position
2. Insert new item at the last position
3. Insert new item before the given value of node
4. Insert new item after the given value of node
5. Search item from the list
6. Display all elements of the list
7. Count total number of items

Reference:

```
Enter the choice:
1
Insert item which you want to inset at first position:
12
Enter the choice:
2
Insert item which you want to insert at last position:
23
Enter the choice:
3
Insert item which you want to insert:
18
Insert item before where you want to insert:
23
Enter the choice:
4
Insert item which you want to insert:
15
Insert item after where you want to insert:
```

```
23
Enter the choice:
5
Insert item which you want to search:
18
Item is available at the 2 position of the list
Enter the choice:
6
List is:
12
18
23
15
Enter the choice:
7
Total number of items are:
4
```

SOLUTION

```
#include <stdio.h>
#include <stdlib.h>

struct node
{
   int data;
   struct node *next;
};
struct node *head;

void insert_F()
{
         printf("\n+++++| INSERT FIRST |+++++\n");
         struct node *ptr;
   int item;
   ptr = (struct node *) malloc(sizeof(struct node *));
   if(ptr == NULL)
   {
      printf("\nOVERFLOW");
   }
   else
   {
      printf("\nEnter value you want to insert: ");
```

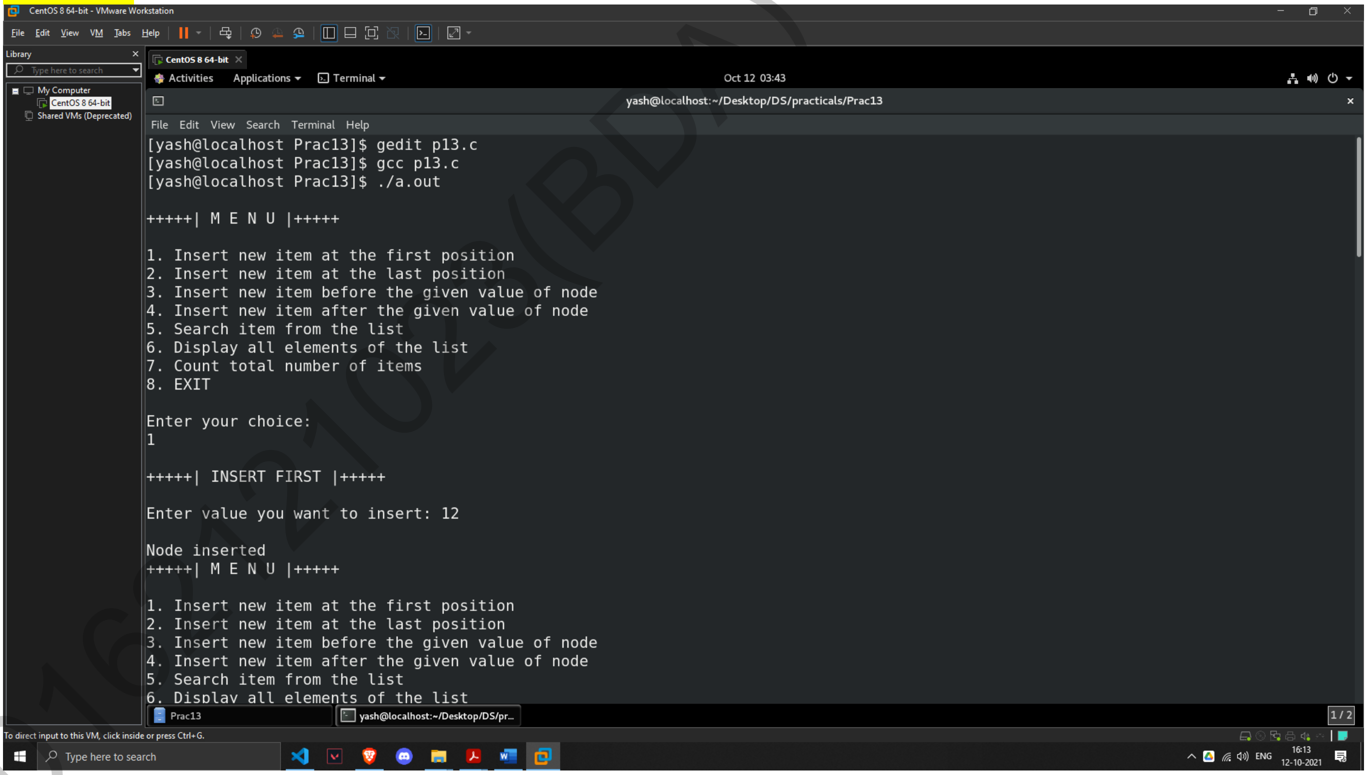
```
        scanf("%d",&item);
        ptr->data = item;
        ptr->next = head;
        head = ptr;
        printf("\nNode inserted");
    }
}
void insert_L()
{
            printf("\n+++++| INSERT LAST |+++++\n");
            struct node *ptr,*temp;
    int item;
    ptr = (struct node*)malloc(sizeof(struct node));
    if(ptr == NULL)
    {
        printf("\nOVERFLOW");
    }
    else
    {
        printf("\nEnter value you want to insert: ");
        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 insert_B()
```

```
{
        printf("\n+++++| INSERT BEFORE |+++++\n");
        int yash, prajapati;
        struct node *new_node, *ptr, *previous;
        printf("Insert item which you want to insert: ");
        scanf("%d", &yash);
        printf("Insert item before where you want to insert: ");
        scanf("%d", &prajapati);
        new_node = (struct node *)malloc(sizeof(struct node));
        new_node->data = yash;
        ptr = head;
        while (ptr->data != prajapati)
        {
                previous = ptr;
                ptr = ptr->next;
        }
        previous->next = new_node;
        new_node->next = ptr;
}

void insert_A()
{
        printf("\n+++++| INSERT AFTER |+++++\n");
        int yash,prajapati;
   struct node *new_node,*ptr,*after;
   printf("Insert item which you want to insert: ");
   scanf("%d",&yash);
   printf("Insert item after where you want to insert: ");
   scanf("%d",&prajapati);
   new_node=(struct node *)malloc(sizeof(struct node));
   new_node->data = yash;
   ptr = head;

   while(ptr->data != prajapati)
   {
      ptr = ptr->next;
   }
   new_node->next = ptr->next;
        ptr->next = new_node;
}

void search()
{
        int YASH,pos=0;
```

```
    struct node *ptr;
    printf("\nInsert item which you want to search: ");
    scanf("%d",&YASH);
    ptr=head;
    while(ptr->next!=NULL)
    {
        if(ptr->data==YASH)
        {
            printf("\nItem is available at the %d position of the list",pos+1);
                                  break;
        }
        else
        {
            pos++;
            ptr=ptr->next;
        }
    }
}

void display()
{
            struct node *ptr;
    ptr = head;
    if(ptr == NULL)
    {
        printf("Empty !!!");
    }
    else
    {
        printf("\n+++++| D I S P L A Y |+++++\n\n");
        while (ptr!=NULL)
        {
            printf("%d ",ptr->data);
            ptr = ptr -> next;
        }
                        printf("\n");
    }
}

void count()
{
            int count=1;
    struct node *ptr;
    ptr=head;
```

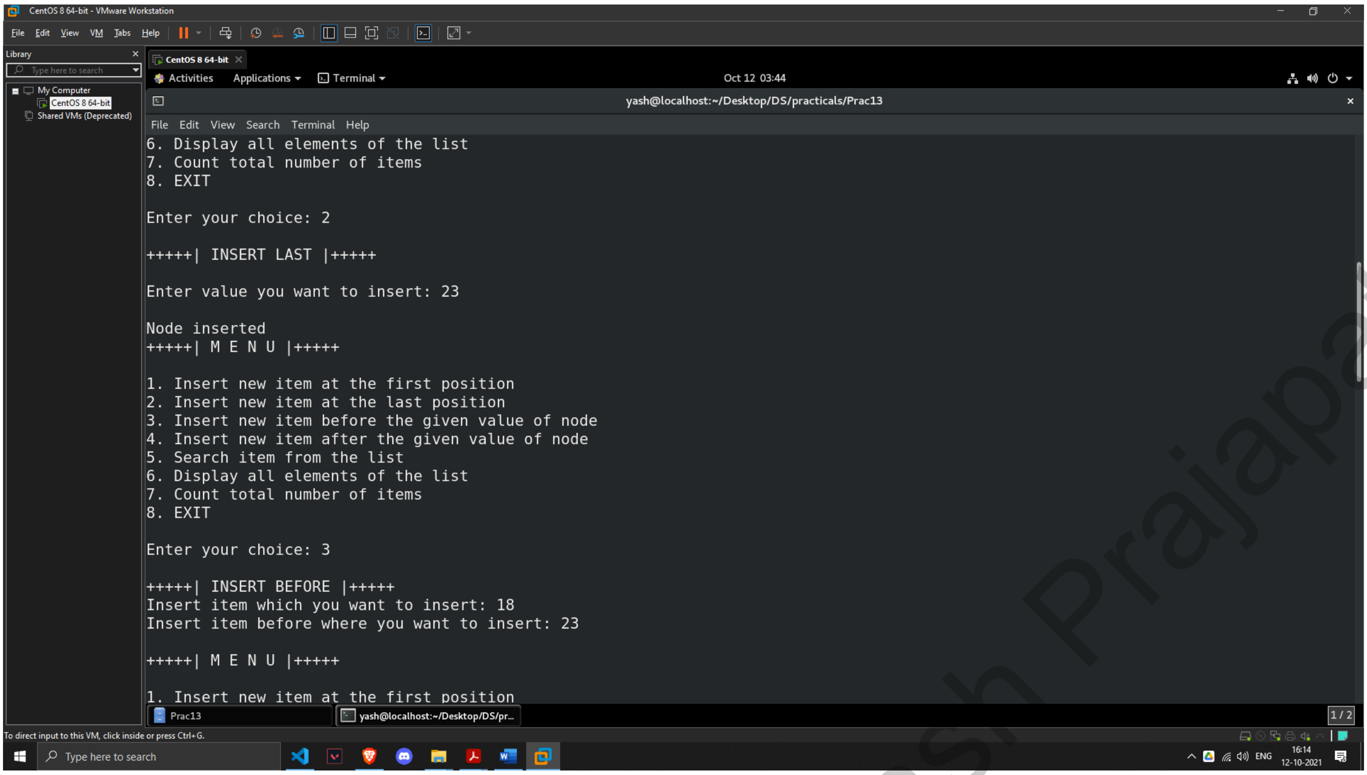
```
   while(ptr->next!=NULL)
   {
      count++;
      ptr=ptr->next;
   }
   printf("Total number of items are:\n%d",count);
}

int main()
{
        int ch;
        while (1)
        {
                RETRY:
                printf("\n+++++| M E N U |+++++\n");
                printf("\n1. Insert new item at the first position");
                printf("\n2. Insert new item at the last position");
                printf("\n3. Insert new item before the given value of node");
                printf("\n4. Insert new item after the given value of node");
                printf("\n5. Search item from the list");
                printf("\n6. Display all elements of the list");
                printf("\n7. Count total number of items");
                printf("\n8. EXIT");
                printf("\n\nEnter your choice: ");
                scanf("%d",&ch);

                switch (ch)
                {
                case 1:
                        insert_F();
                        break;
                case 2:
                        insert_L();
                        break;
                case 3:
                        insert_B();
                        break;
                case 4:
                        insert_A();
                        break;
                case 5:
                        search();
                        break;
                case 6:
                    display();
                    break;
          case 7:
                    count();
                    break;
          case 8:
                    printf("\n+++++| EXITING SYSTEM |+++++\n");
                    exit(0);
                    break;

          default:
          printf("\nINVALID CHOICE !! Please try again.....\n");
          goto RETRY;
                    break;
          }
}

return 0;
}
```

OUTPUT

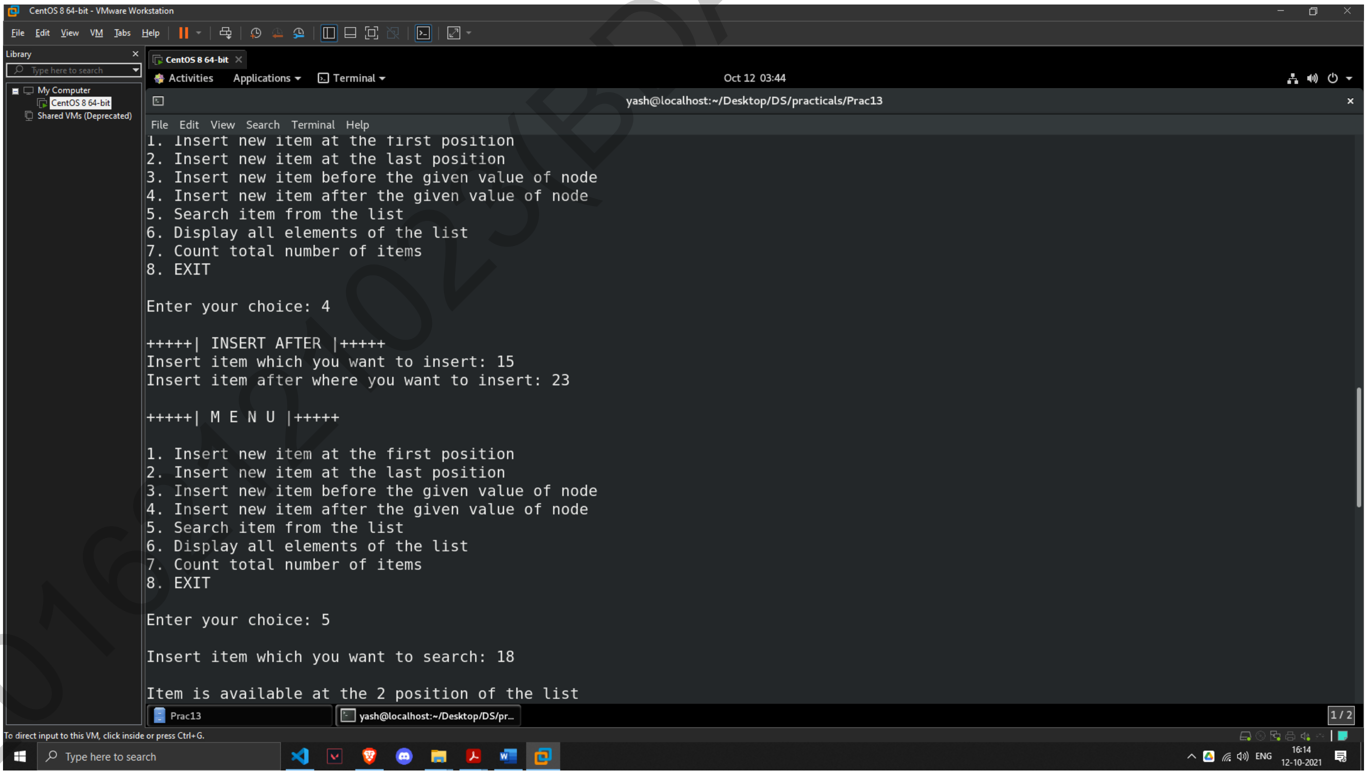


CentOS 8 64-bit - VMware Workstation
File Edit View VM Tabs Help
Library
My Computer
CentOS 8 64-bit
Shared VMs (Deprecated)
Activities Applications Terminal
Oct 12 03:43
yash@localhost:~/Desktop/DS/practicals/Prac13
File Edit View Search Terminal Help
Prac13
1 / 2
To direct input to this VM, click inside or press Ctrl+G.
Type here to search
ENG
16:13
12-10-2021

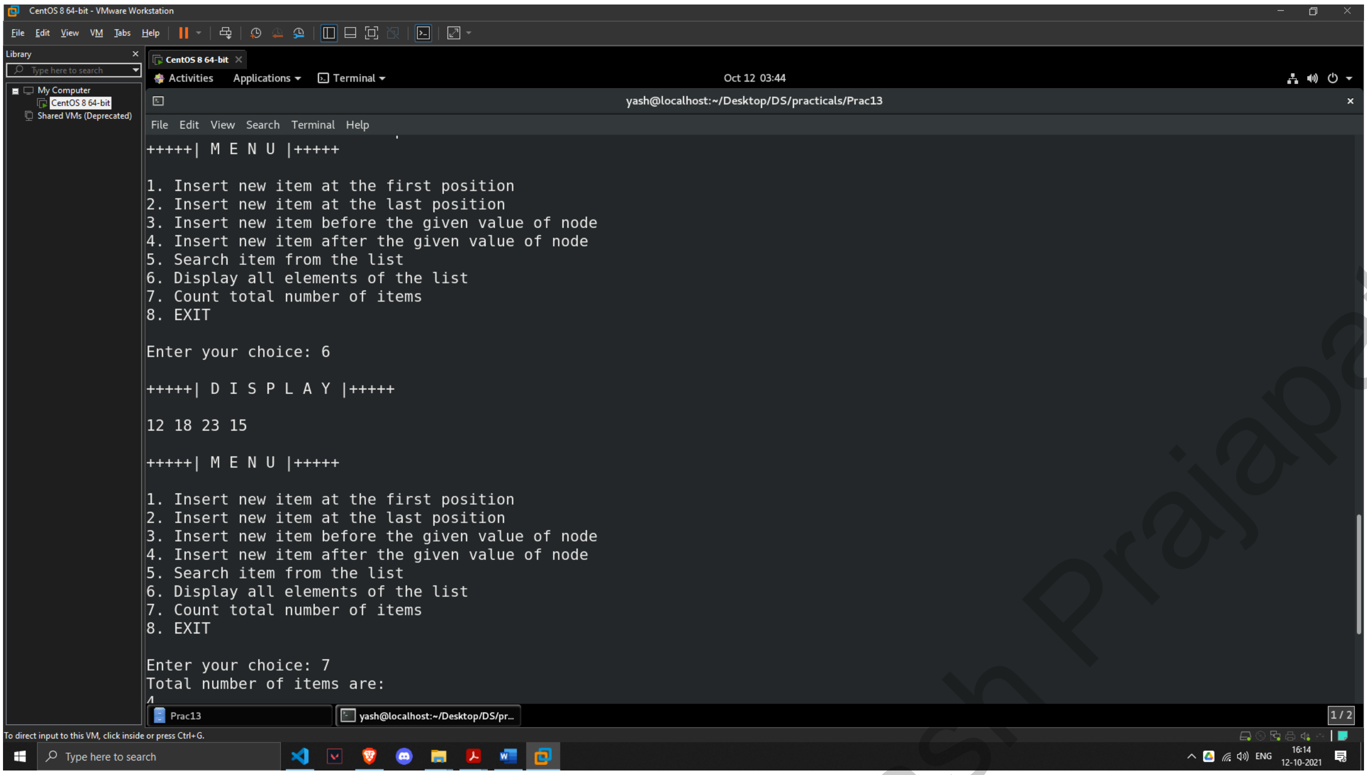
```
[yash@localhost Prac13]$ gedit p13.c
[yash@localhost Prac13]$ gcc p13.c
[yash@localhost Prac13]$ ./a.out

+++++| M E N U |+++++

1. Insert new item at the first position
2. Insert new item at the last position
3. Insert new item before the given value of node
4. Insert new item after the given value of node
5. Search item from the list
6. Display all elements of the list
7. Count total number of items
8. EXIT

Enter your choice:
1

+++++| INSERT FIRST |+++++

Enter value you want to insert: 12

Node inserted
+++++| M E N U |+++++

1. Insert new item at the first position
2. Insert new item at the last position
3. Insert new item before the given value of node
4. Insert new item after the given value of node
5. Search item from the list
6. Display all elements of the list
```


CentOS 8 64-bit - VMware Workstation
Activities
Applications
Terminal
Oct 12 03:44
yash@localhost:~/Desktop/DS/practicals/Prac13
File Edit View Search Terminal Help
Prac13
1 / 2
16:14
12-10-2021

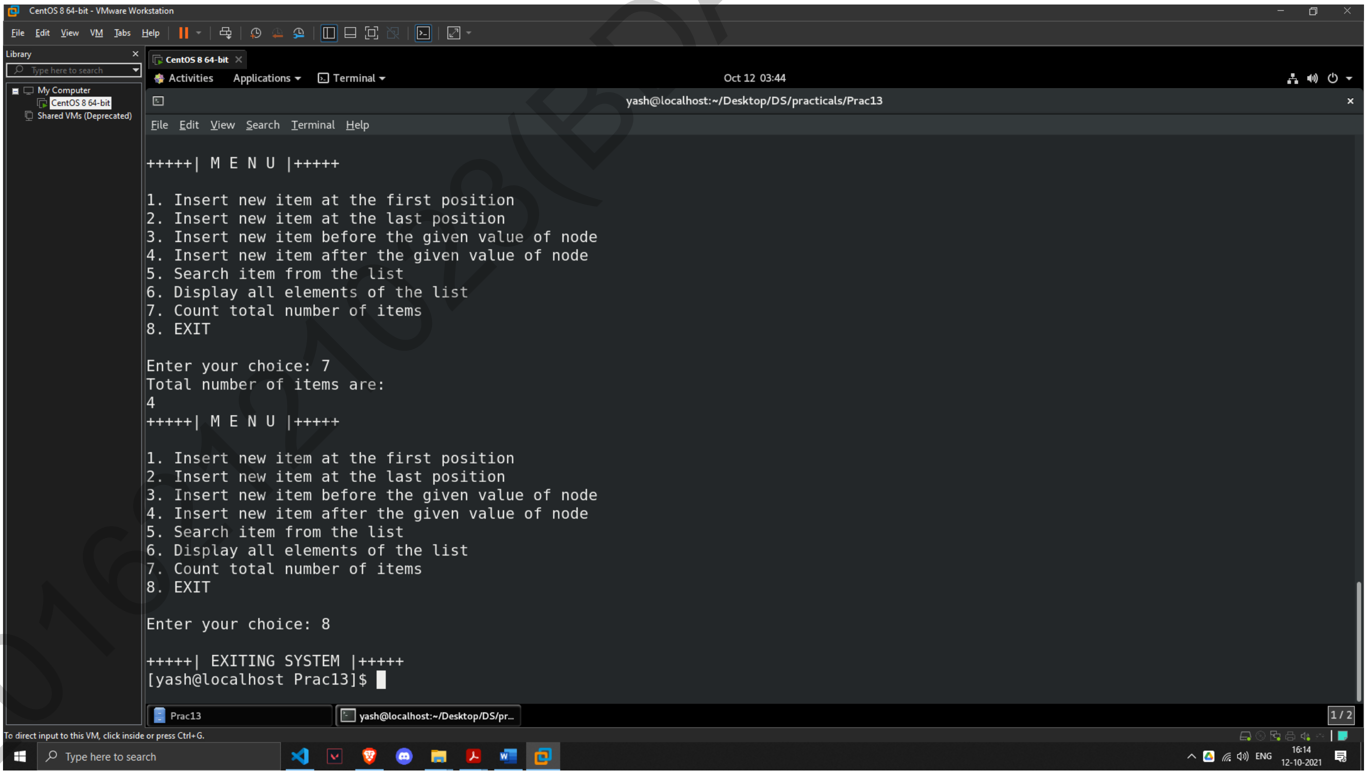
```
6. Display all elements of the list
7. Count total number of items
8. EXIT

Enter your choice: 2

+++++| INSERT LAST |+++++

Enter value you want to insert: 23

Node inserted
+++++| M E N U |+++++

1. Insert new item at the first position
2. Insert new item at the last position
3. Insert new item before the given value of node
4. Insert new item after the given value of node
5. Search item from the list
6. Display all elements of the list
7. Count total number of items
8. EXIT

Enter your choice: 3

+++++| INSERT BEFORE |+++++
Insert item which you want to insert: 18
Insert item before where you want to insert: 23

+++++| M E N U |+++++

1. Insert new item at the first position
```


CentOS 8 64-bit - VMware Workstation
Activities
Applications
Terminal
Oct 12 03:44
yash@localhost:~/Desktop/DS/practicals/Prac13
File Edit View Search Terminal Help
Prac13
1 / 2
16:14
12-10-2021

```
1. Insert new item at the first position
2. Insert new item at the last position
3. Insert new item before the given value of node
4. Insert new item after the given value of node
5. Search item from the list
6. Display all elements of the list
7. Count total number of items
8. EXIT

Enter your choice: 4

+++++| INSERT AFTER |+++++
Insert item which you want to insert: 15
Insert item after where you want to insert: 23

+++++| M E N U |+++++

1. Insert new item at the first position
2. Insert new item at the last position
3. Insert new item before the given value of node
4. Insert new item after the given value of node
5. Search item from the list
6. Display all elements of the list
7. Count total number of items
8. EXIT

Enter your choice: 5

Insert item which you want to search: 18

Item is available at the 2 position of the list
```


CentOS 8 64-bit - VMware Workstation
Activities
Applications
Terminal
Oct 12 03:44
yash@localhost:~/Desktop/DS/practicals/Prac13
Prac13
1 / 2

```
+++++| M E N U |+++++

1. Insert new item at the first position
2. Insert new item at the last position
3. Insert new item before the given value of node
4. Insert new item after the given value of node
5. Search item from the list
6. Display all elements of the list
7. Count total number of items
8. EXIT

Enter your choice: 6

+++++| D I S P L A Y |+++++

12 18 23 15

+++++| M E N U |+++++

1. Insert new item at the first position
2. Insert new item at the last position
3. Insert new item before the given value of node
4. Insert new item after the given value of node
5. Search item from the list
6. Display all elements of the list
7. Count total number of items
8. EXIT

Enter your choice: 7
Total number of items are:
4
```


CentOS 8 64-bit - VMware Workstation
Activities
Applications
Terminal
Oct 12 03:44
yash@localhost:~/Desktop/DS/practicals/Prac13
Prac13
1 / 2

```
+++++| M E N U |+++++

1. Insert new item at the first position
2. Insert new item at the last position
3. Insert new item before the given value of node
4. Insert new item after the given value of node
5. Search item from the list
6. Display all elements of the list
7. Count total number of items
8. EXIT

Enter your choice: 7
Total number of items are:
4
+++++| M E N U |+++++

1. Insert new item at the first position
2. Insert new item at the last position
3. Insert new item before the given value of node
4. Insert new item after the given value of node
5. Search item from the list
6. Display all elements of the list
7. Count total number of items
8. EXIT

Enter your choice: 8

+++++| EXITING SYSTEM |+++++
[yash@localhost Prac13]$
```