Name : Saransh Gautam

Branch : IT

Roll no. : 11913036

Subject : Data Structure

ACKNOWLEDGEMENT

The success and final outcome of this project required a lot of guidance and assistance from many people and I am extremely privileged to have got this ass along the completion of my project. All that I have done is only due to such supervision and assistance and I wound not forget to thank them.

I respect and thank Mrs. Rama Chawla, for providing me an opportunity to do the project work and giving us all support and guidance, which made me complete the project duly. I am extremely thankful to her for providing such a nice support and guidance, although she had busy schedule managing the corporate affairs.

Experiment VI(v)

Aim: Swapping Elements in pair.

Code:

```
#include<stdio.h>
#include<stdlib.h>
#include<conio.h>
struct node
{
        int data;
        struct node *next;
};
struct node *head, *newnode, *temp;
int count=0;
void create_list();
void print_list();
void swap_pair();
main()
{
        int i;
        head=NULL;
        printf("Enter the choice for corresponding operations\n");
        printf("Press 1 to create list\n");
        printf("Press 2 to print list\n");
        printf("Press 3 to swap pair of elements\n");
        printf("Press 0 to exit this program\n");
```

```
        scanf("%d" ,&i);
        while(i)
        {
                switch(i)
                {
                        case 1: create_list();
                                        break;
                        case 2: print_list();
                                        break;
                        case 3: swap_pair();
                                        printf("The elements are swapped in pair\n");
                                        break;
                }
                printf("Enter the choice\n");
                scanf("%d" ,&i);
        }
}
void create_list()
{
        int i, n;
        while(1)
        {
                printf("Press 1 if you wanna insert element\nElse if you completed
creating list press 0\n");
                scanf("%d" ,&i);
                if (i==1)
                {
```

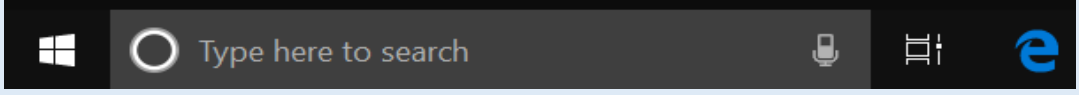
```
                newnode=(struct node *)malloc(sizeof (struct node));
                printf("Enter the value you wanna insert\n");
                scanf("%d" ,&(newnode->data));
                newnode->next=NULL;
                if (head==NULL)
                {
                        head=newnode;
                        temp=newnode;
                }
                else
                {
                        temp->next=newnode;
                        temp=newnode;
                }
                count++;
        }
        else if (i==0)
                break;
    }
}
void print_list()
{
    temp=head;
    while (temp!=0)
    {
        printf("%d\t" ,temp->data);
```

```
            temp=temp->next;
        }
        printf("\n");
}
void swap_pair()
{
        int a;
        temp=head;
        if (count%2 != 0)
        {
            while(temp->next!=NULL)
            {
                a=temp->data;
                temp->data=temp->next->data;
                temp->next->data=a;
                temp=temp->next->next;
            }
        }
        else
        {
            while(temp!=NULL)
            {
                a=temp->data;
                temp->data=temp->next->data;
                temp->next->data=a;
                temp=temp->next->next;
```

```
            }
        }
    }
```

For even number of nodes:

```
Enter the choice for corresponding operations
Press 1 to create list
Press 2 to print list
Press 3 to swap pair of elements
Press 0 to exit this program
1
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
10
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
20
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
30
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
40
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
50
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
60
Press 1 if you wanna insert element
Else if you completed creating list press 0
0
Enter the choice
2
10      20      30      40      50      60
Enter the choice
3
The elements are swapped in pair
Enter the choice
2
20      10      40      30      60      50
Enter the choice
```



For odd number of nodes:

```
C:\Saransh\Second sem\Data Structure\Codes\Singly linked list\Experiment 6.5.exe
Enter the choice for corresponding operations
Press 1 to create list
Press 2 to print list
Press 3 to swap pair of elements
Press 0 to exit this program
1
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
25
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
50
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
75
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
100
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
125
Press 1 if you wanna insert element
Else if you completed creating list press 0
0
Enter the choice
2
25      50      75      100     125
Enter the choice
3
The elements are swapped in pair
Enter the choice
2
50      25      100     75      125
Enter the choice
0
```

Experiment VI(vi.a)

Aim: Finding second last node

Code:

```
#include<stdio.h>
#include<stdlib.h>
#include<conio.h>
struct node
{
        int data;
        struct node *next;
};
struct node *head, *newnode, *temp;
int count=0;
void create_list();
void print_list();
void find_seclast();
main()
{
        head=NULL;
        int i;
        printf("Enter the choice for the corresponding task\n");
        printf("Press 1 to create linked list\n");
        printf("Press 2 to print linked list\n");
        printf("Press 3 to find the second last element of linked list\n");
        printf("Press 0 to exit this program\n");
```

```
        scanf("%d" ,&i);
        while (i)
        {
                switch(i)
                {
                        case 1: create_list();
                                        break;
                        case 2: print_list();
                                        break;
                        case 3: find_seclast();
                                        break;
                }
                printf("Enter your choice\n");
                scanf("%d" ,&i);
        }
}
void create_list()
{
        int i, n;
        while(1)
        {
                printf("Press 1 if you wanna insert element\nElse if you completed
creating list press 0\n");
                scanf("%d" ,&i);
                if (i==1)
                {
                        newnode=(struct node *)malloc(sizeof (struct node));
```

```
                printf("Enter the value you wanna insert\n");
                scanf("%d" ,&(newnode->data));
                newnode->next=NULL;
                if (head==NULL)
                {
                        head=newnode;
                        temp=newnode;
                }
                else
                {
                        temp->next=newnode;
                        temp=newnode;
                }
                count++;
        }
        else if (i==0)
                break;
    }
}
void print_list()
{
    temp=head;
    while (temp!=0)
    {
        printf("%d\t" ,temp->data);
        temp=temp->next;
```

```
        }
        printf("\n");
}
void find_seclast()
{
        int i;
        temp=head;
        i=1;
        while(i<count-1)
        {
                temp=temp->next;
                i++;
        }
        printf("The second last element is %d\n" ,temp->data);
}
```

```
Enter the choice for the corresponding task
Press 1 to create linked list
Press 2 to print linked list
Press 3 to find the second last element of linked list
Press 0 to exit this program
1
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
10
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
20
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
30
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
40
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
50
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
60
Press 1 if you wanna insert element
Else if you completed creating list press 0
0
Enter your choice
2
10      20      30      40      50      60
Enter your choice
3
The second last element is 50
Enter your choice
0

--------------------------------
Process exited after 16.28 seconds with return value 0
```

Experiment VI(vi.b)

Aim: Concatenate two linked lists in ascending order.

Code:

```
#include<stdio.h>
#include<stdlib.h>
#include<conio.h>
struct node
{
        int data;
        struct node *next;
};
void create_list(struct node **head);
void print_list(struct node *head);
void bub_sort(struct node *head);
void concatenate(struct node **head1, struct node **head2);
void swap(struct node *a, struct node *b);
main()
{
        struct node *head1, *head2;
        head1=NULL;
        head2=NULL;
        printf("Enter the first linked list\n");
        create_list(&head1);
        printf("Enter the second linked list\n");
```

```
	create_list(&head2);
	bub_sort(head1);
	bub_sort(head2);
	printf("Both the linked list are sorted and are as follows :\n");
	print_list(head1);
	print_list(head2);
	concatenate(&head1, &head2);
	print_list(head1);
}
void create_list(struct node **head)
{
	struct node *newnode, *temp;
	int i, n;
	while(1)
	{
		printf("Press 1 if you wanna insert element\nElse if you completed
creating list press 0\n");
		scanf("%d" ,&i);
		if (i==1)
		{
			newnode=(struct node *)malloc(sizeof (struct node));
			printf("Enter the value you wanna insert\n");
			scanf("%d" ,&(newnode->data));
			newnode->next=NULL;
			if (*head==NULL)
			{
				*head=newnode;
```

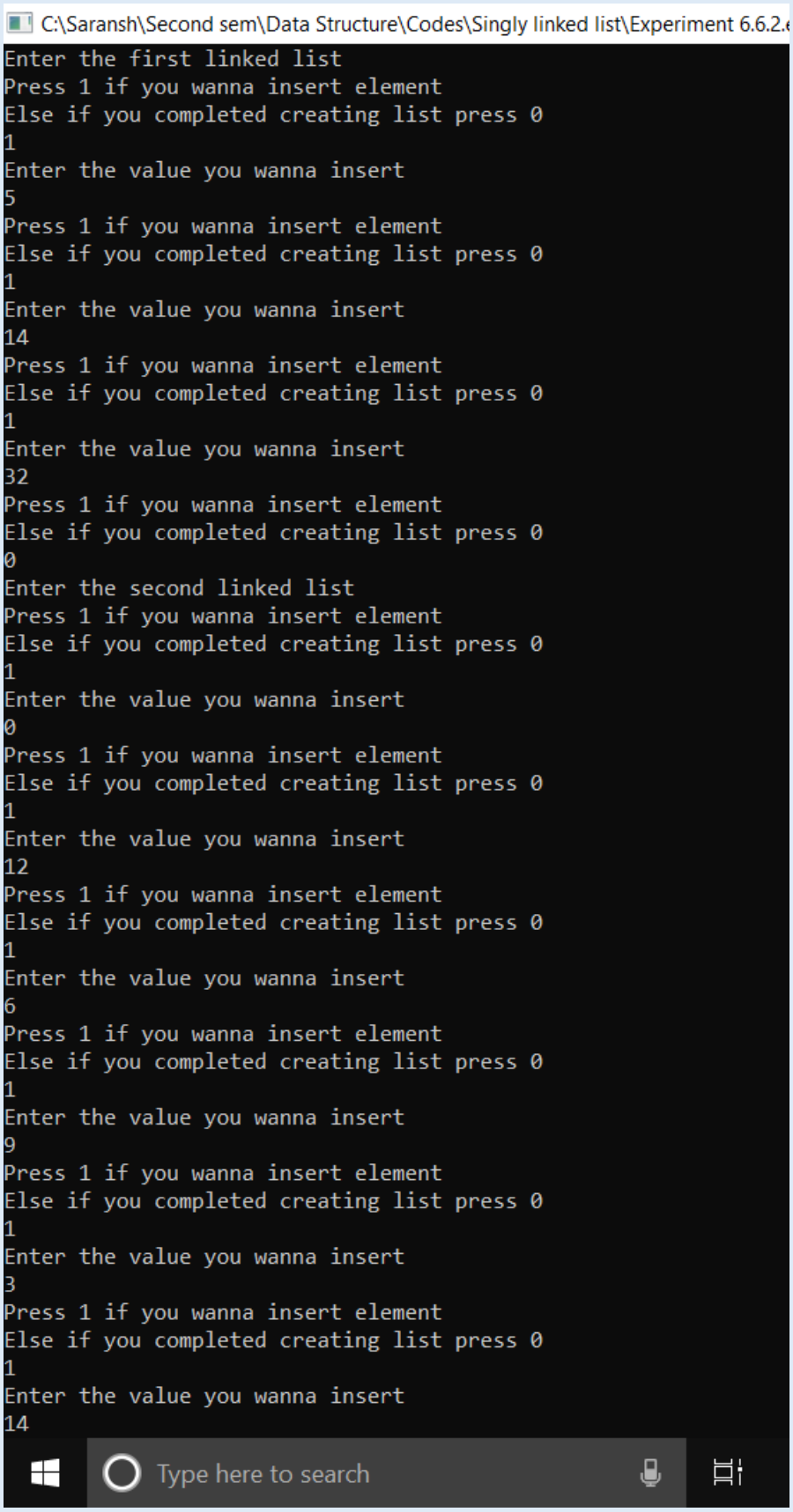
```
                        temp=newnode;
                }
                else
                {
                        temp->next=newnode;
                        temp=newnode;
                }
                //count++;
        }
        else if (i==0)
                break;
    }

}
void print_list( struct node *head)
{
    struct node *temp;
    temp=head;
    while (temp!=0)
    {
        printf("%d\t" ,temp->data);
        temp=temp->next;
    }
    printf("\n");
}
void bub_sort(struct node *head)
```

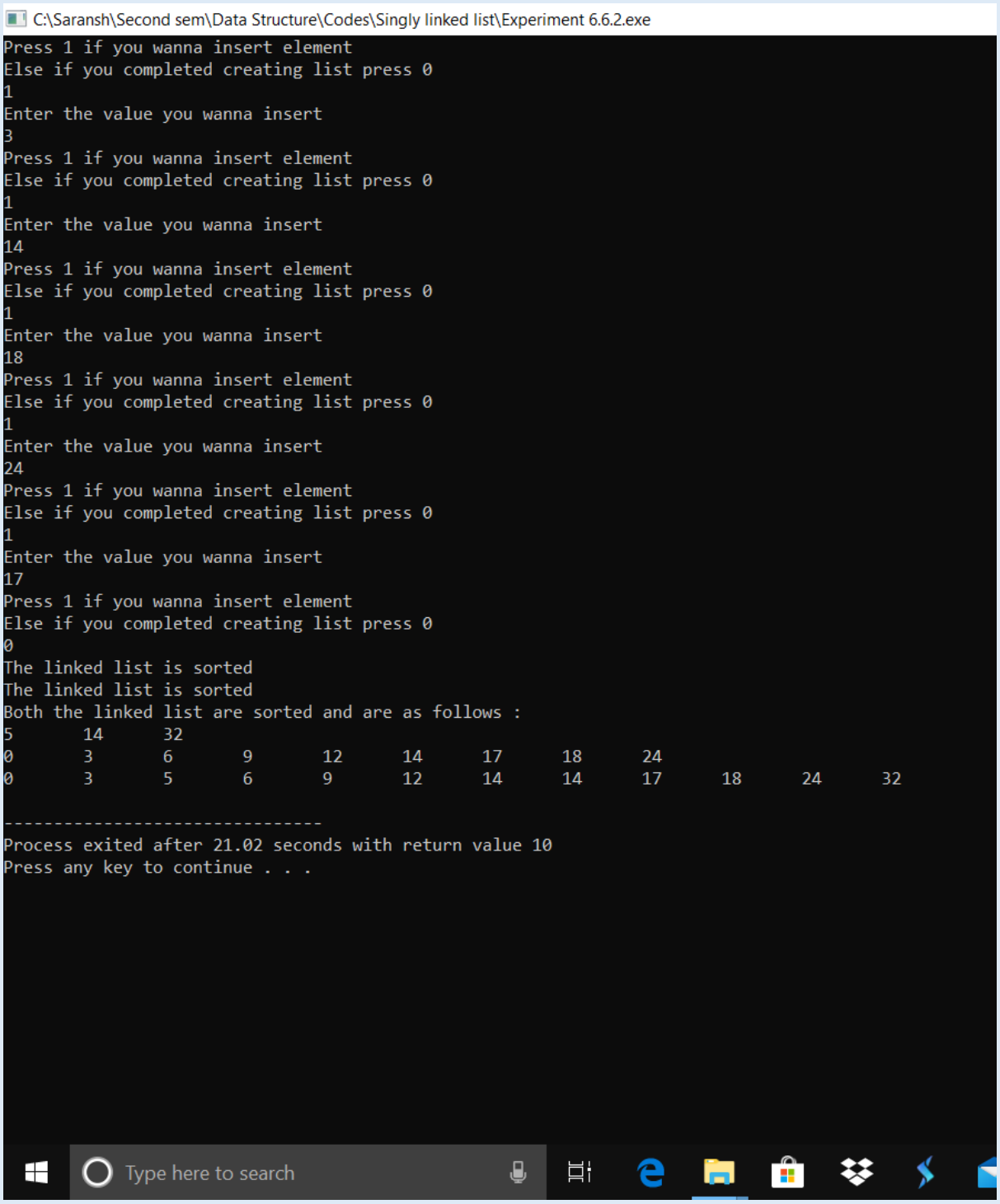
```
{
        struct node *temp , *ptr;
        int flag=0;
        temp=head;
        do
        {
                temp=head;
                flag=0;
                while(temp->next!=NULL)
                {
                        if (temp->data > temp->next->data)
                        {
                                swap(temp, temp->next);
                                flag=1;
                        }
                        temp=temp->next;
                }
        }
        while(flag);
        printf("The linked list is sorted\n");
}
void concatenate(struct node **head1, struct node **head2)
{
        struct node *temp1, *temp2, *pre, *newnode;
        temp1=*head1;
        temp2=*head2;
```

```
        pre=*head1;
        while(temp2!=NULL)
        {
                //temp1=head1;
                //pre=head1;
                while(temp1!=NULL)
                {
                        if(temp1->data >= temp2->data)
                        {
                                if(pre->data <= temp2->data)
                                {
                                        newnode=(struct node *)malloc(sizeof (struct
node));
                                        newnode->data=temp2->data;
                                        pre->next=newnode;
                                        newnode->next=temp1;
                                        pre=newnode;
                                        break;
                                }
                                else if (pre->data > temp2->data)
                                {
                                        newnode=(struct node *)malloc(sizeof (struct
node));
                                        newnode->data=temp2->data;
                                        newnode->next=temp1;
                                        *head1=newnode;
                                        pre=*head1;
```

```
                                        break;


                                }
                        }
                        if(temp1 != *head1)
                        {
                                pre=pre->next;
                        }
                        temp1=temp1->next;
                }
                if (temp1 == NULL && temp2!=NULL)
                {
                        newnode=(struct node *)malloc(sizeof (struct node));
                        newnode->data = temp2->data;
                        newnode->next=NULL;
                        pre->next = newnode;
                        pre=newnode;


                }
                temp2=temp2->next;


        }

}
void swap(struct node *a, struct node *b)
{
```

```
        int temp;
        temp=a->data;
        a->data=b->data;
        b->data=temp;
}
```


C:\Saransh\Second sem\Data Structure\Codes\Singly linked list\Experiment 6.6.2.
Enter the first linked list
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
5
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
14
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
32
Press 1 if you wanna insert element
Else if you completed creating list press 0
0
Enter the second linked list
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
0
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
12
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
6
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
9
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
3
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
14
Type here to search


C:\Saransh\Second sem\Data Structure\Codes\Singly linked list\Experiment 6.6.2.exe
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
3
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
14
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
18
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
24
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
17
Press 1 if you wanna insert element
Else if you completed creating list press 0
0
The linked list is sorted
The linked list is sorted
Both the linked list are sorted and are as follows :
5 14 32
0 3 6 9 12 14 17 18 24
0 3 5 6 9 12 14 14 17 18 24 32

Process exited after 21.02 seconds with return value 10
Press any key to continue . . .
Type here to search

Experiment VI(vi.c)

Aim: Sort singly linked list.

Code:

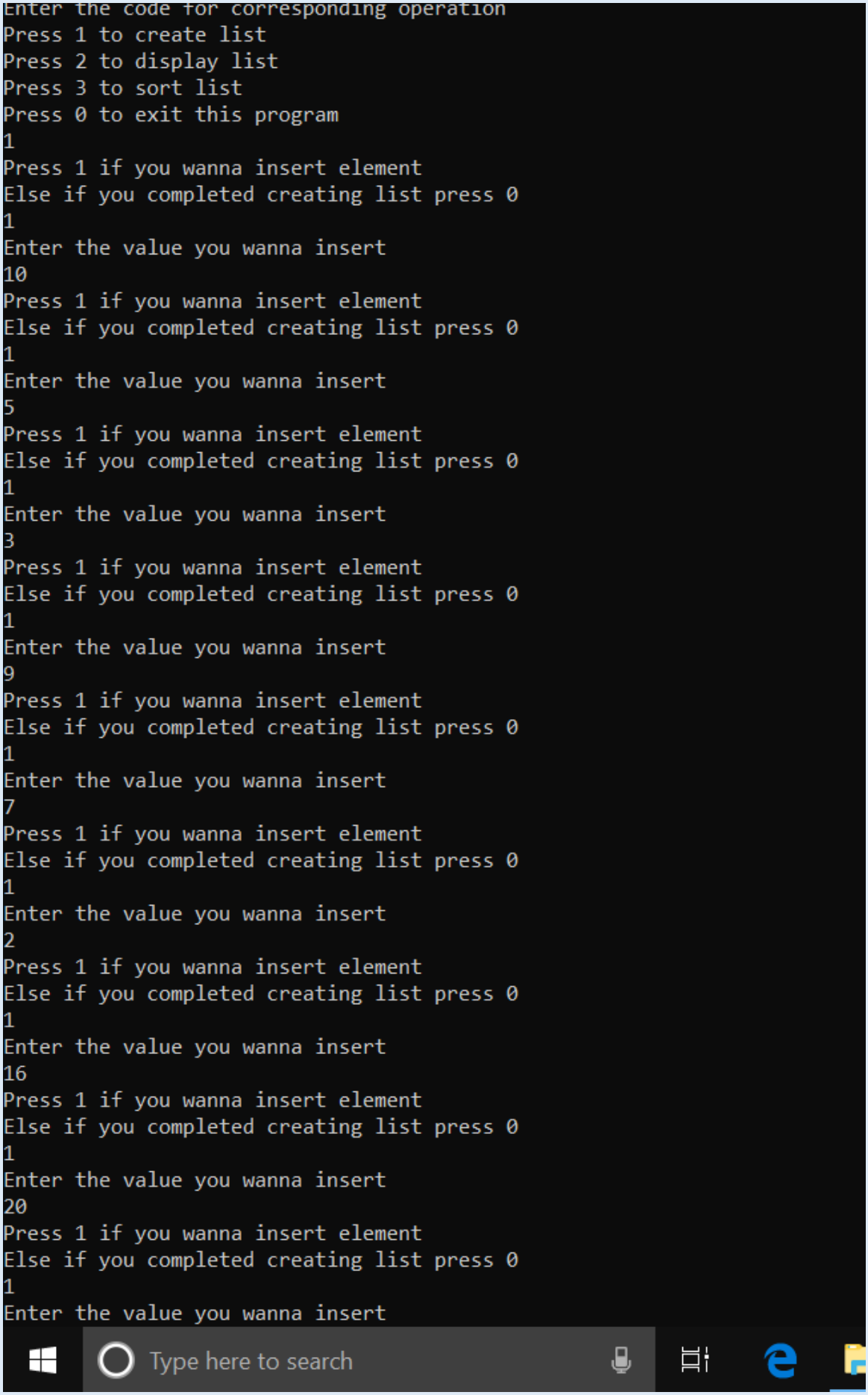
```
#include<stdio.h>
#include<stdlib.h>
#include<conio.h>
struct node
{
        int data;
        struct node *next;
};
void create_list(struct node **head);
void print_list(struct node *head);
void bub_sort(struct node *head);
void swap(struct node *a, struct node *b);
main()
{
        struct node *head;
        int i;
        head=NULL;
        printf("Enter the code for corresponding operation\n");
        printf("Press 1 to create list\n");
        printf("Press 2 to display list\n");
        printf("Press 3 to sort list\n");
        printf("Press 0 to exit this program\n");
```

```
        scanf("%d" ,&i);
        while(i)
        {
                switch(i)
                {
                        case 1: create_list(&head);
                                        break;
                        case 2: print_list(head);
                                        break;
                        case 3: bub_sort(head);
                                        break;
                }
                printf("Enter the choice\n");
                scanf("%d" ,&i);
        }
}
void create_list(struct node **head)
{
        struct node *newnode, *temp;
        int i, n;
        while(1)
        {
                printf("Press 1 if you wanna insert element\nElse if you completed
creating list press 0\n");
                scanf("%d" ,&i);
                if (i==1)
                {
```

```
                newnode=(struct node *)malloc(sizeof (struct node));
                printf("Enter the value you wanna insert\n");
                scanf("%d" ,&(newnode->data));
                newnode->next=NULL;
                if (*head==NULL)
                {
                        *head=newnode;
                        temp=newnode;
                }
                else
                {
                        temp->next=newnode;
                        temp=newnode;
                }
                //count++;
        }
        else if (i==0)
                break;
}

}
void print_list( struct node *head)
{
        struct node *temp;
        temp=head;
        while (temp!=0)
```

```
        {
                printf("%d\t" ,temp->data);
                temp=temp->next;
        }
        printf("\n");
}
void bub_sort(struct node *head)
{
        struct node *temp , *ptr;
        int flag=0;
        temp=head;
        do
        {
                temp=head;
                flag=0;
                while(temp->next!=NULL)
                {
                        if (temp->data > temp->next->data)
                        {
                                swap(temp, temp->next);
                                flag=1;
                        }
                        temp=temp->next;
                }
        }
        while(flag);
```

```
        printf("The linked list is sorted\n");
}
void swap(struct node *a, struct node *b)
{
        int temp;
        temp=a->data;
        a->data=b->data;
        b->data=temp;
}
```


Enter the code for corresponding operation
Press 1 to create list
Press 2 to display list
Press 3 to sort list
Press 0 to exit this program
1
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
10
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
5
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
3
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
9
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
7
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
2
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
16
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
20
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
Type here to search

```
Enter the value you wanna insert
20
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
13
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
22
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
7
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
19
Press 1 if you wanna insert element
Else if you completed creating list press 0
0
Enter the choice
2
10      5       3       9       7       2       16      20      13      22      7       19
Enter the choice
3
The linked list is sorted
Enter the choice
2
2       3       5       7       7       9       10      13      16      19      20      22
Enter the choice
0

--------------------------------
Process exited after 71.74 seconds with return value 0
Press any key to continue . . .
```

Experiment VI(vii.a)

Aim: Print alternate nodes from list.

Code:

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

```
#include<stdlib.h>
#include<conio.h>
struct node
{
        int data;
        struct node *next;
};
struct node *head, *newnode, *temp;
int count=0;
void create_list();
void print_list();
void printf_alt();
main()
{
        head=NULL;
        int i;
        printf("Enter the choice for the corresponding task\n");
        printf("Press 1 to create list\n");
        printf("Press 2 to print whole list\n");
        printf("Press 3 to print alternate elements\n");
        printf("Press 0 tp exit this program\n");
        scanf("%d" ,&i);
        while (i)
        {
                switch (i)
                {
```

```
                case 1: create_list();
                        break;
                case 2: print_list();
                        break;
                case 3: printf_alt();
                        break;
            }
            printf("Enter the choice\n");
            scanf("%d" ,&i);
        }
}
void create_list()
{
        int i, n;
        while(1)
        {
            printf("Press 1 if you wanna insert element\nElse if you completed
creating list press 0\n");
            scanf("%d" ,&i);
            if (i==1)
            {
                newnode=(struct node *)malloc(sizeof (struct node));
                printf("Enter the value you wanna insert\n");
                scanf("%d" ,&(newnode->data));
                newnode->next=NULL;
                if (head==NULL)
                {
```

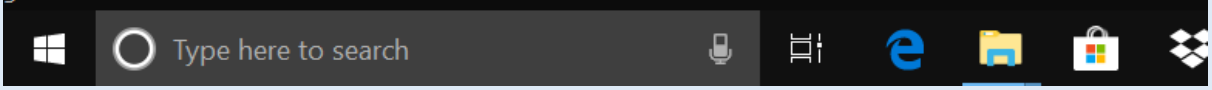
```
                        head=newnode;
                        temp=newnode;
                }
                else
                {
                        temp->next=newnode;
                        temp=newnode;
                }
                count++;
        }
        else if (i==0)
                break;
    }
}
void print_list()
{
    temp=head;
    while (temp!=0)
    {
        printf("%d\t" ,temp->data);
        temp=temp->next;
    }
    printf("\n");
}
void printf_alt()
{
```

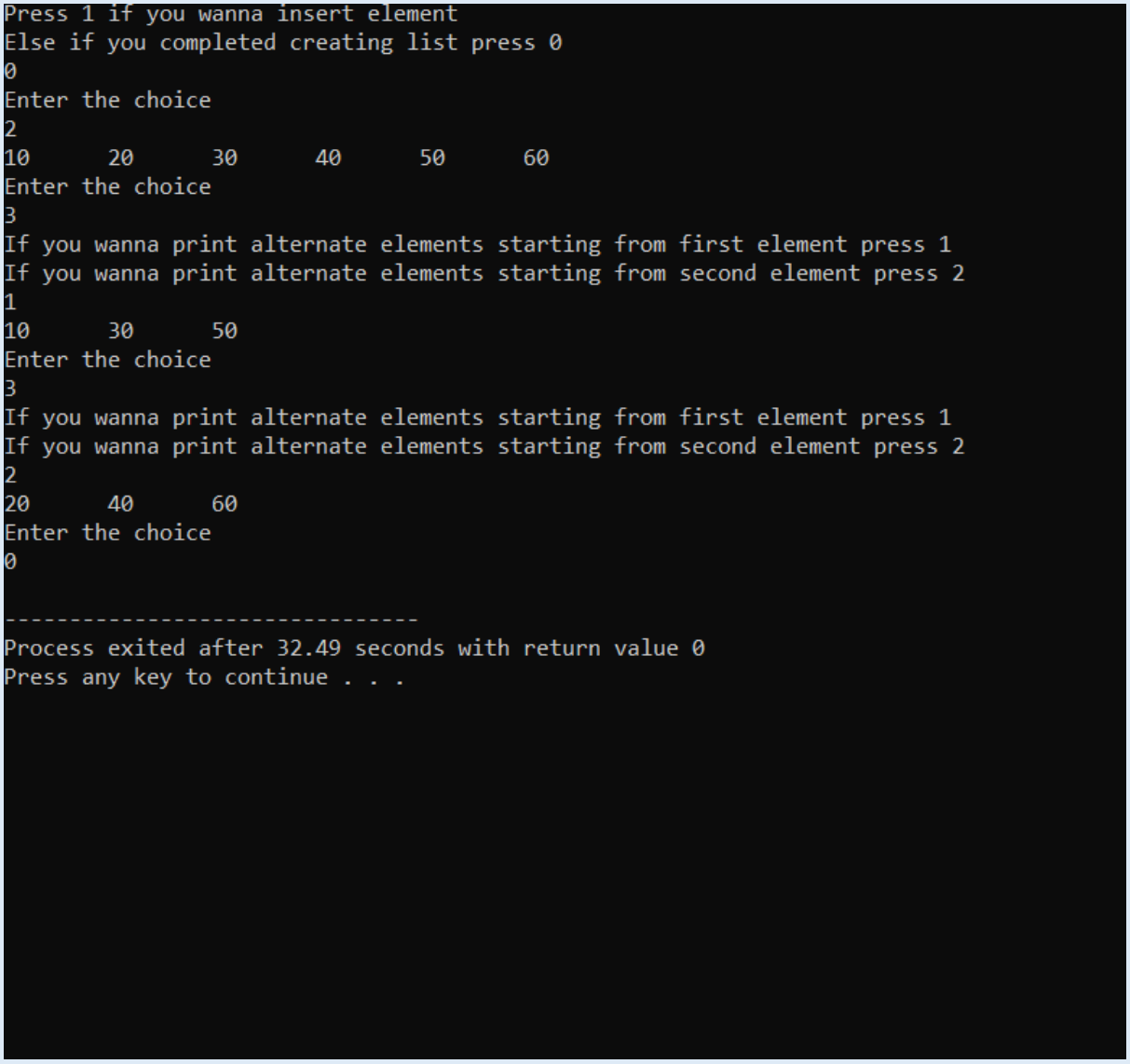
```
temp=head;
int i;
printf("If you wanna print alternate elements starting from first element press 1\n");
printf("If you wanna print alternate elements starting from second element press 2\n");
scanf("%d" ,&i);
if (i==1)
{
	if(count%2 == 0)
	{
		while (1)
		{
			printf("%d\t" ,temp->data);
			temp=temp->next->next;
			if(temp==NULL)
				break;
		}
	}
	else
	{
		while(1)
		{
			printf("%d\t" ,temp->data);
			if(temp->next==NULL)
				break;
			temp=temp->next->next;
```

```
                }
        }
}
else if(i==2)
{
        temp=temp->next;
        if(count%2 == 0)
        {
                while (1)
                {
                        printf("%d\t" ,temp->data);
                        if(temp->next == NULL)
                                break;
                        temp=temp->next->next;

                }
        }
        else
        {
                while(1)
                {
                        printf("%d\t" ,temp->data);
                        temp=temp->next->next;
                        if(temp == NULL)
                                break;
                }
```

```
                }


        }
        printf("\n");
}
```

```
Enter the choice for the corresponding task
Press 1 to create list
Press 2 to print whole list
Press 3 to print alternate elements
Press 0 tp exit this program
1
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
10
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
20
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
30
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
40
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
50
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
60
Press 1 if you wanna insert element
Else if you completed creating list press 0
0
Enter the choice
2
10      20      30      40      50      60
Enter the choice
3
If you wanna print alternate elements starting from first element press 1
If you wanna print alternate elements starting from second element press 2
1
10      30      50
Enter the choice
3
```





Experiment VI(vii.b)

Aim: Concatenate even elements from two linked lists.

Code:

```
#include<stdio.h>

#include<stdlib.h>

#include<conio.h>

struct node

{
```

```
        int data;
        struct node *next;
};
void create_list(struct node **head);
void print_list(struct node *head);
void concatenate(struct node *head1, struct node *head2, struct node
**head3);
main()
{
        struct node *head1, *head2 ,*head3;
        head1=NULL;
        head2=NULL;
        head3=NULL;
        printf("Enter the first list\n");
        create_list(&head1);
        printf("Enter the second list\n");
        create_list(&head2);
        printf("The lists you entered are :\n");
        print_list(head1);
        print_list(head2);
        printf("Now the even positioned elements of the list are
concatenated\n");
        concatenate(head1, head2, &head3);
        printf("The new list is :\n");
        print_list(head3);
}
void create_list(struct node **head)
```

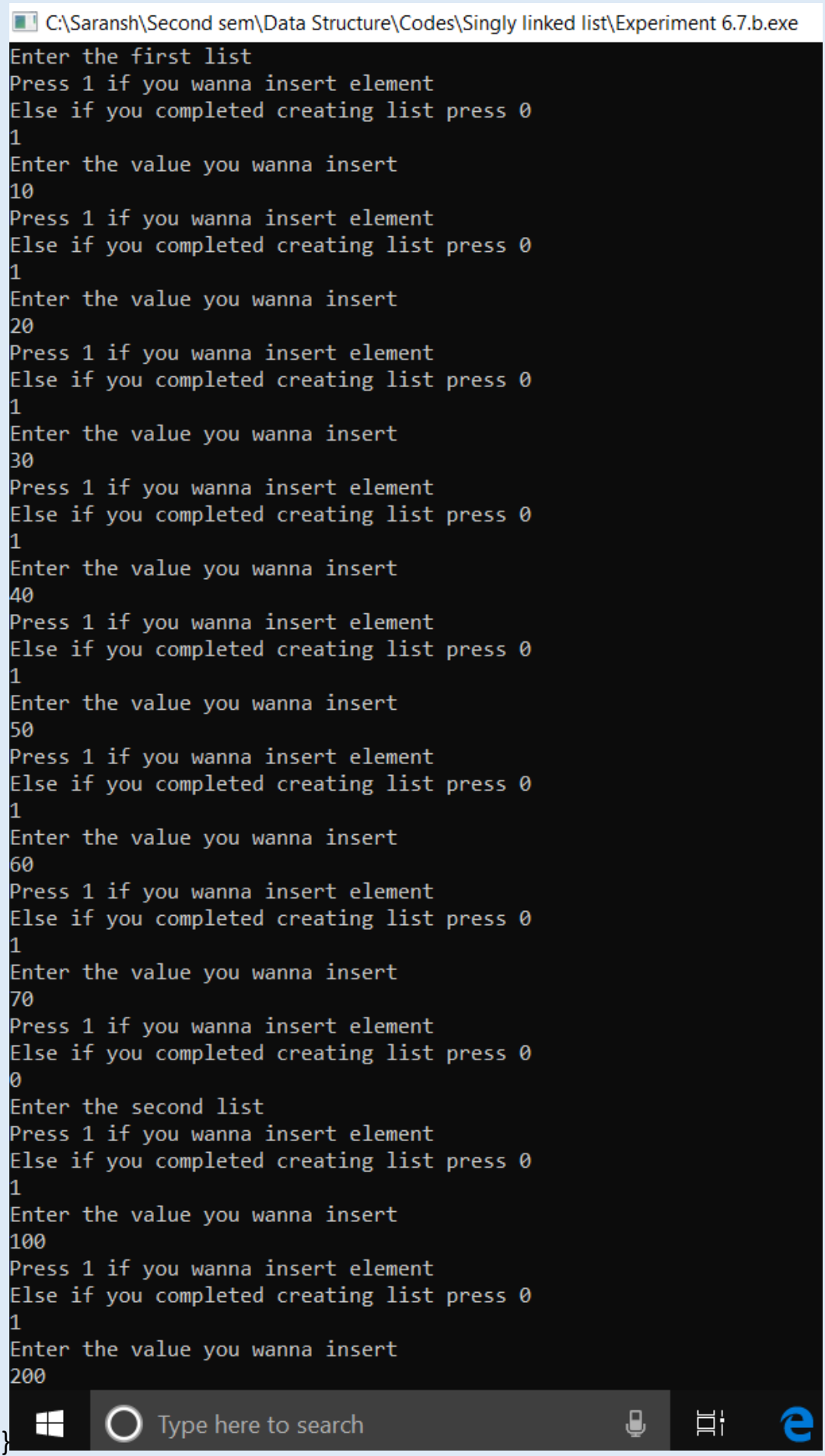
```
{
        struct node *newnode, *temp;
        int i, n;
        while(1)
        {
                printf("Press 1 if you wanna insert element\nElse if you completed creating list press 0\n");
                scanf("%d" ,&i);
                if (i==1)
                {
                        newnode=(struct node *)malloc(sizeof (struct node));
                        printf("Enter the value you wanna insert\n");
                        scanf("%d" ,&(newnode->data));
                        newnode->next=NULL;
                        if (*head==NULL)
                        {
                                *head=newnode;
                                temp=newnode;
                        }
                        else
                        {
                                temp->next=newnode;
                                temp=newnode;
                        }
                        //count++;
                }
                else if (i==0)
```

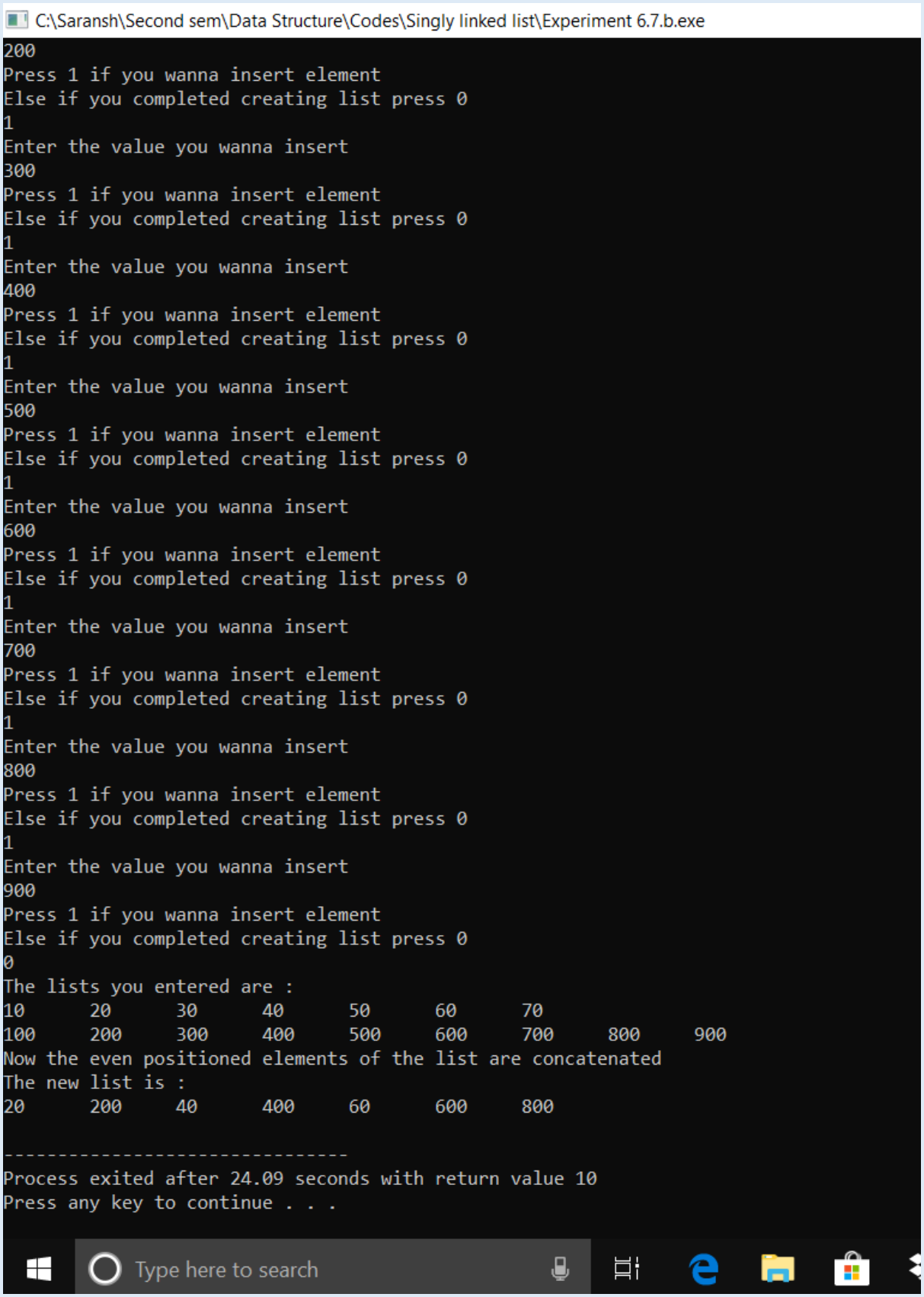
```
                        break;
        }

}
void print_list( struct node *head)
{
        struct node *temp;
        temp=head;
        while (temp!=0)
        {
                printf("%d\t" ,temp->data);
                temp=temp->next;
        }
        printf("\n");
}
void concatenate(struct node *head1, struct node *head2, struct node
**head3)
{
        struct node *temp1, *temp2, *newnode, *temp3;
        temp1=head1;
        temp2=head2;
        temp1=temp1->next;
        temp2=temp2->next;
        while(temp1!=NULL && temp2!=NULL)
        {
                newnode=(struct node *)malloc(sizeof (struct node));
                newnode->data=temp1->data;
```

```
                newnode->next=NULL;
                if(*head3 == NULL)
                {
                        *head3=newnode;
                        temp3=newnode;
                }
                else
                {
                        temp3->next=newnode;
                        temp3=newnode;
                }
                newnode=(struct node *)malloc (sizeof (struct node));
                newnode->data=temp2->data;
                newnode->next=NULL;
                temp3->next=newnode;
                temp3=newnode;
                if(temp1->next==NULL || temp2->next==NULL)
                {
                        break;
                }
                temp1=temp1->next->next;
                temp2=temp2->next->next;
        }
        if((temp1==NULL || temp1->next==NULL) && (temp2!=NULL || temp2-
>next!=NULL))
        {
                while(temp2!=NULL)
```

```
            {
                newnode=(struct node *)malloc(sizeof (struct node));
                newnode->data=temp2->data;
                newnode->next=NULL;
                temp3->next=newnode;
                temp3=newnode;
                if(temp2->next==NULL)
                {
                    break;
                }
                temp2=temp2->next->next;
            }

        }
        else if ((temp2==NULL || temp2->next==NULL) && (temp1!=NULL ||
temp1->next!=NULL))
        {
            while (temp1!=NULL)
            {
                newnode=(struct node *)malloc(sizeof (struct node));
                newnode->data=temp1->data;
                newnode->next=NULL;
                temp3->next=newnode;
                temp3=newnode;
                if(temp1->next==NULL)
                {
                    break;
```

```
                }
                temp1=temp1->next->next;
        }
}
```


C:\Saransh\Second sem\Data Structure\Codes\Singly linked list\Experiment 6.7.b.exe
Enter the first list
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
10
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
20
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
30
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
40
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
50
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
60
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
70
Press 1 if you wanna insert element
Else if you completed creating list press 0
0
Enter the second list
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
100
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
200
Type here to search
}


C:\Saransh\Second sem\Data Structure\Codes\Singly linked list\Experiment 6.7.b.exe
200
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
300
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
400
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
500
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
600
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
700
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
800
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
900
Press 1 if you wanna insert element
Else if you completed creating list press 0
0
The lists you entered are :
10 20 30 40 50 60 70
100 200 300 400 500 600 700 800 900
Now the even positioned elements of the list are concatenated
The new list is :
20 200 40 400 60 600 800

Process exited after 24.09 seconds with return value 10
Press any key to continue . . .
Type here to search

Experiment VI(vii.c)

Aim: Finding first common element among two strings.

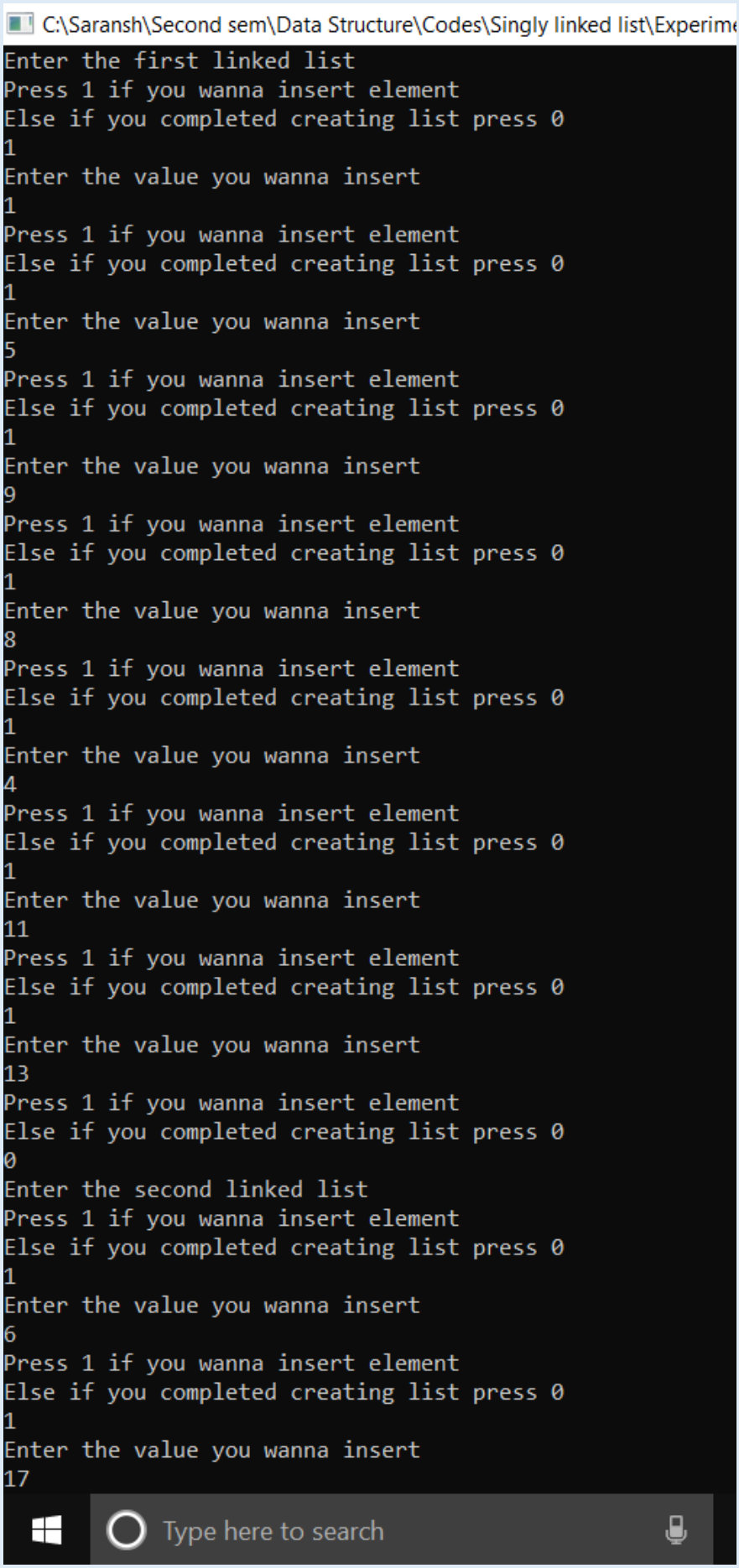
Code:

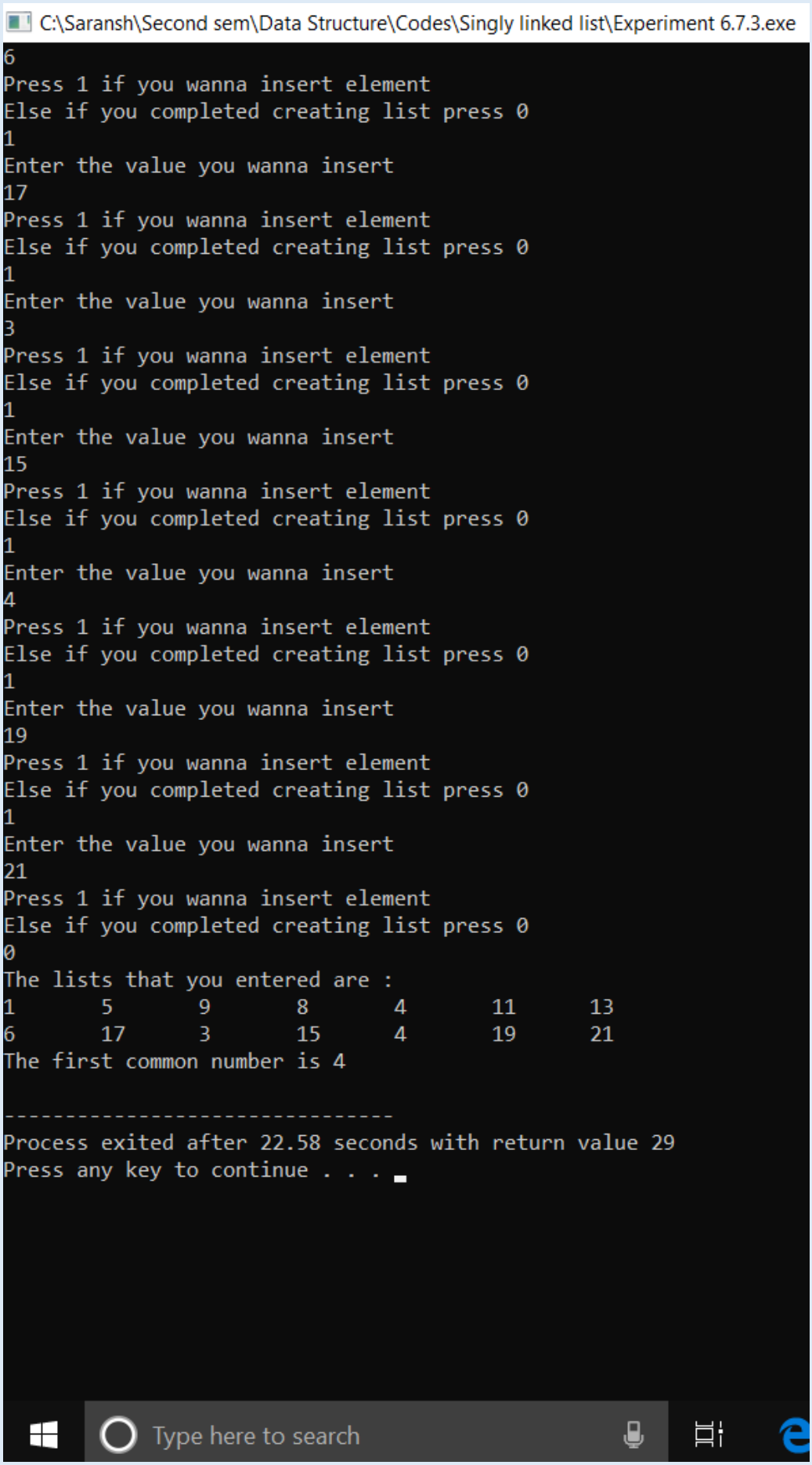
```
#include<stdio.h>
#include<stdlib.h>
#include<conio.h>
struct node
{
        int data;
        struct node *next;
};
void create_list(struct node **head);
void print_list(struct node *head);
void chk_fst(struct node *head1, struct node *head2);
main()
{
        struct node *head1, *head2;
        head1=NULL;
        head2=NULL;
        printf("Enter the first linked list\n");
        create_list(&head1);
        printf("Enter the second linked list\n");
        create_list(&head2);
        printf("The lists that you entered are :\n");
```

```
	print_list(head1);
	print_list(head2);
	chk_fst(head1, head2);
}
void create_list(struct node **head)
{
	struct node *newnode, *temp;
	int i, n;
	while(1)
	{
		printf("Press 1 if you wanna insert element\nElse if you completed creating list press 0\n");
		scanf("%d" ,&i);
		if (i==1)
		{
			newnode=(struct node *)malloc(sizeof (struct node));
			printf("Enter the value you wanna insert\n");
			scanf("%d" ,&(newnode->data));
			newnode->next=NULL;
			if (*head==NULL)
			{
				*head=newnode;
				temp=newnode;
			}
			else
			{
				temp->next=newnode;
```

```
                        temp=newnode;
                }
                //count++;
        }
        else if (i==0)
                break;
    }

}
void print_list( struct node *head)
{
    struct node *temp;
    temp=head;
    while (temp!=0)
    {
        printf("%d\t" ,temp->data);
        temp=temp->next;
    }
    printf("\n");
}
void chk_fst(struct node *head1, struct node *head2)
{
    struct node *temp1, *temp2;
    temp1=head1;
    temp2=head2;
    int flag=0;
```

```
        while (temp1!=NULL)
        {
                temp2=head2;
                while(temp2!=NULL)
                {
                        if (temp2->data == temp1->data)
                        {
                                printf("The first common number is %d\n" ,temp1-
>data);
                                flag=1;
                                break;
                        }
                        temp2=temp2->next;
                }
                if(flag==1)
                        break;
                temp1=temp1->next;
        }
        if(flag==0)
        {
                printf("The are no common elements in the two list\n");
        }
}
```


C:\Saransh\Second sem\Data Structure\Codes\Singly linked list\Experime
Enter the first linked list
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
1
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
5
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
9
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
8
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
4
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
11
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
13
Press 1 if you wanna insert element
Else if you completed creating list press 0
0
Enter the second linked list
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
6
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
17
Type here to search


C:\Saransh\Second sem\Data Structure\Codes\Singly linked list\Experiment 6.7.3.exe
6
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
17
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
3
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
15
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
4
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
19
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
21
Press 1 if you wanna insert element
Else if you completed creating list press 0
0
The lists that you entered are :
1 5 9 8 4 11 13
6 17 3 15 4 19 21
The first common number is 4

Process exited after 22.58 seconds with return value 29
Press any key to continue . . .
Type here to search

Experiment VI(vii.d)

Aim: Finding number of occurrence of all elements in linked list.

Code:

```
#include<stdio.h>
#include<stdlib.h>
#include<conio.h>
struct node
{
        int data;
        struct node *next;
};
struct supp
{
        int check;
        int count;
        struct supp *after;
};
struct supp *head_s, *temp_s, *newnode_s, *pre;
struct node *head, *newnode, *temp;
void create_list();
void print_list();
void rep_chk();
void print_rep();
```

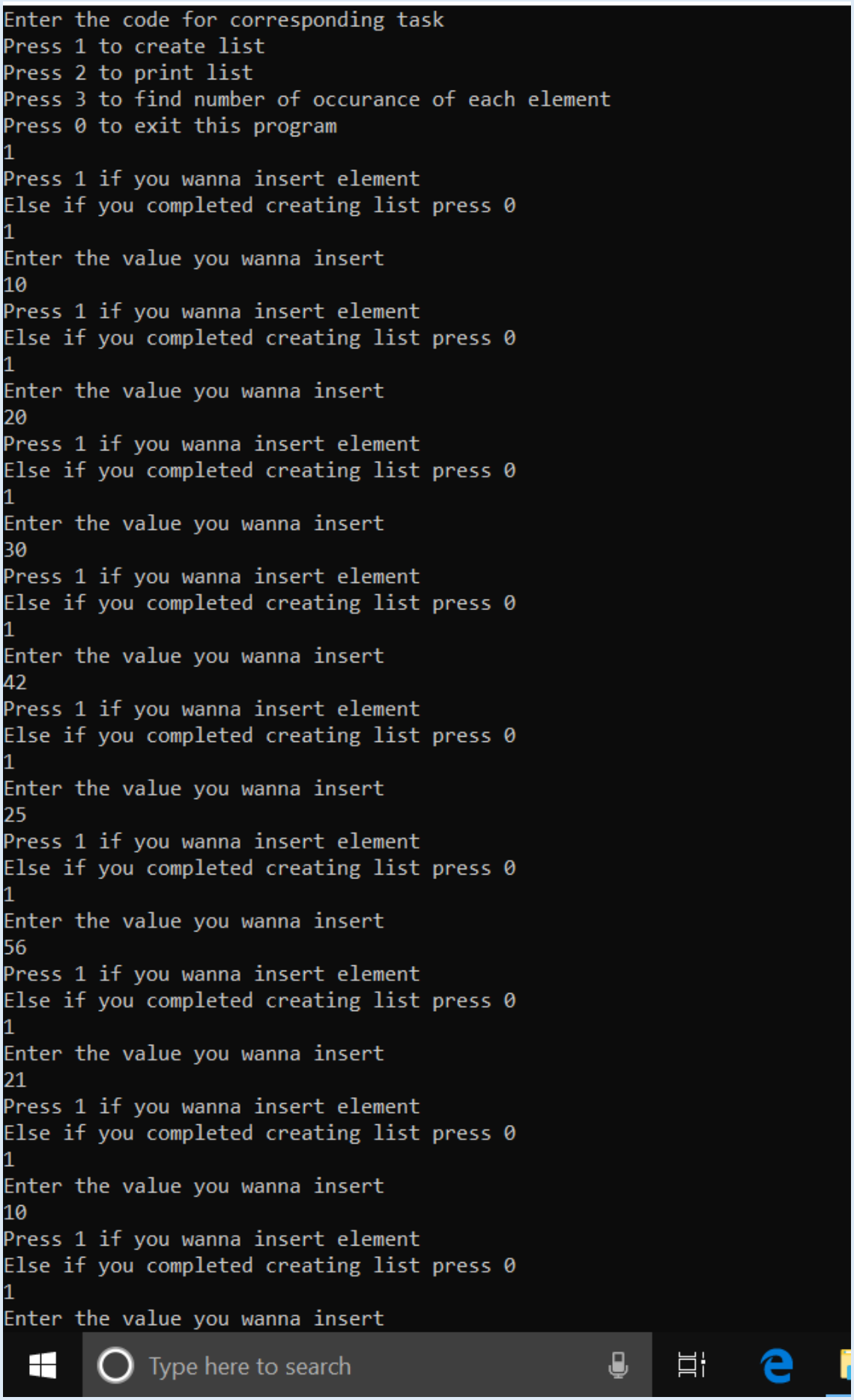
```
main()
{
        head=NULL;
        head_s=NULL;
        int i;
        printf("Enter the code for corresponding task\n");
        printf("Press 1 to create list\n");
        printf("Press 2 to print list\n");
        printf("Press 3 to find number of occurance of each element\n");
        printf("Press 0 to exit this program\n");
        scanf("%d" ,&i);
        while(i)
        {
                switch (i)
                {
                        case 1: create_list();
                                        break;
                        case 2: print_list();
                                        break;
                        case 3: rep_chk();
                                        break;
                }
                printf("Enter the choice\n");
                scanf("%d" ,&i);
        }
}
```

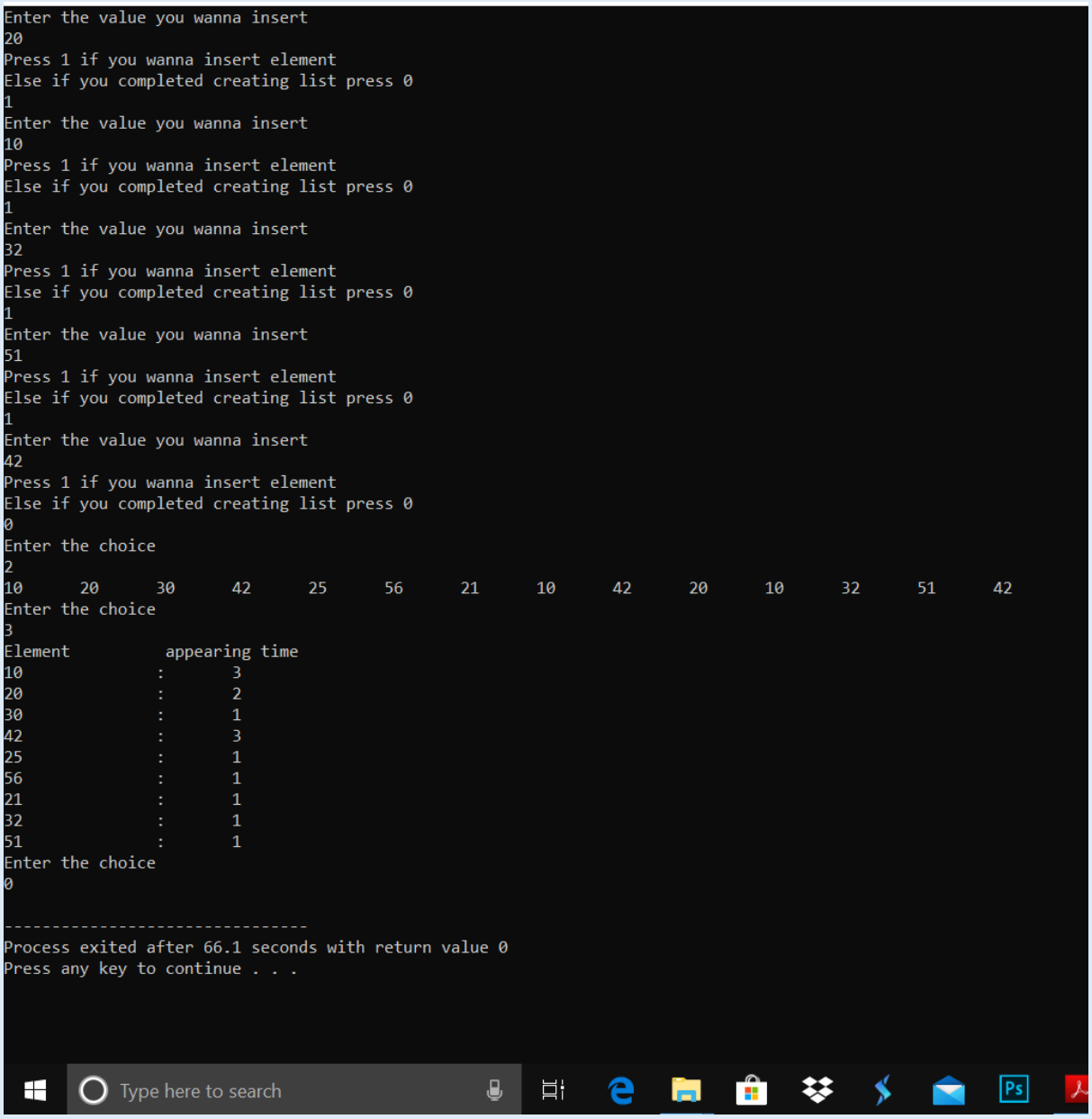
```
void create_list()
{
	int i, n;
	while(1)
	{
		printf("Press 1 if you wanna insert element\nElse if you completed creating list press 0\n");
		scanf("%d" ,&i);
		if (i==1)
		{
			newnode=(struct node *)malloc(sizeof (struct node));
			printf("Enter the value you wanna insert\n");
			scanf("%d" ,&(newnode->data));
			newnode->next=NULL;
			if (head==NULL)
			{
				head=newnode;
				temp=newnode;
			}
			else
			{
				temp->next=newnode;
				temp=newnode;
			}
			//count++;
		}
		else if (i==0)
```

```
                        break;
        }
}
void print_list()
{
        temp=head;
        while (temp!=0)
        {
                printf("%d\t" ,temp->data);
                temp=temp->next;
        }
        printf("\n");
}
void rep_chk()
{
        int flag=0;
        temp=head;
        temp_s=head_s;
        while(temp!=NULL)
        {
                temp_s=head_s;
                flag=0;
                while(temp_s!=NULL)
                {
                        if (temp->data == temp_s->check)
                        {
```

```
                flag=1;
                break;
        }
        temp_s=temp_s->after;
}
if (flag==1)
{
        temp_s->count++;
}
else
{
        newnode_s=(struct supp *)malloc(sizeof (struct supp));
        newnode_s->after=NULL;
        newnode_s->check=temp->data;
        newnode_s->count=1;
        if (head_s == NULL)
        {
                head_s=newnode_s;
                pre=newnode_s;
        }
        else
        {
                pre->after=newnode_s;
                pre=newnode_s;
        }
}
```

```
                temp=temp->next;
        }
        print_rep();

}
void print_rep()
{
        temp_s=head_s;
        printf ("Element \t appearing time\n");
        while(temp_s!=NULL)
        {
                printf("%d\t:\t%d\n" ,temp_s->check, temp_s->count);
                temp_s=temp_s->after;
        }
}
```


Enter the code for corresponding task
Press 1 to create list
Press 2 to print list
Press 3 to find number of occurance of each element
Press 0 to exit this program
1
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
10
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
20
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
30
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
42
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
25
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
56
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
21
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
10
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
Type here to search


Enter the value you wanna insert
20
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
10
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
32
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
51
Press 1 if you wanna insert element
Else if you completed creating list press 0
1
Enter the value you wanna insert
42
Press 1 if you wanna insert element
Else if you completed creating list press 0
0
Enter the choice
2
10 20 30 42 25 56 21 10 42 20 10 32 51 42
Enter the choice
3
Element appearing time
10 : 3
20 : 2
30 : 1
42 : 3
25 : 1
56 : 1
21 : 1
32 : 1
51 : 1
Enter the choice
0
Process exited after 66.1 seconds with return value 0
Press any key to continue . . .
Type here to search