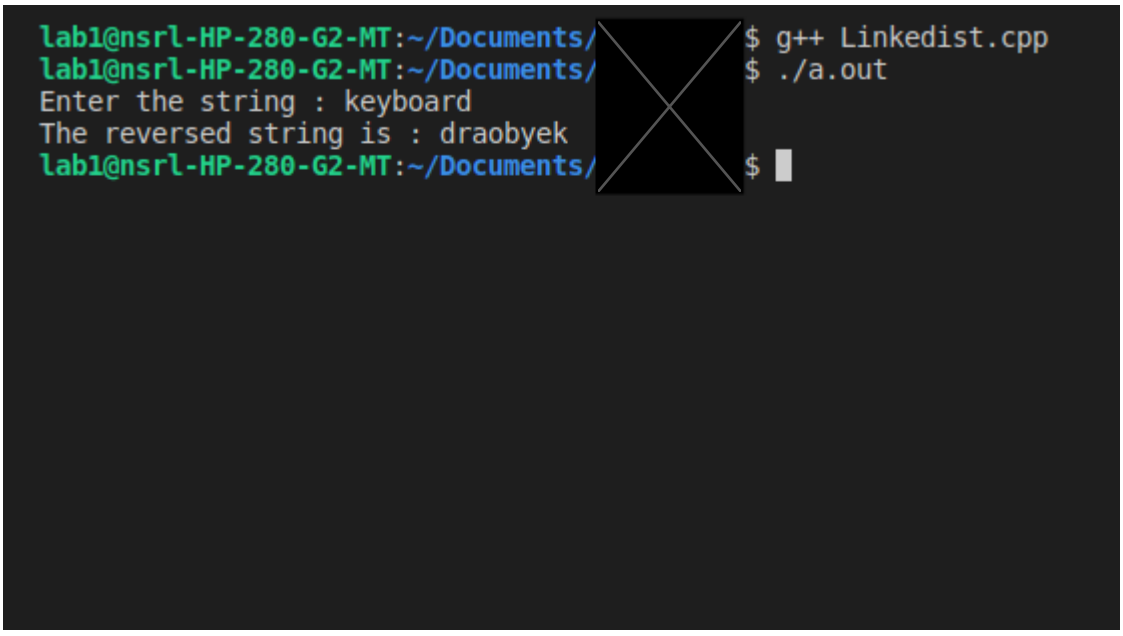
```
/*
//Q1
#include<iostream>
using namespace std;

class Node
{
	public :
		char data;
		Node *next;
};

Node* reverse(Node *start, Node *start2)
{
	if(start->next == NULL)
	{
		return(start);
	}
	else
	{
		start2 = reverse(start->next, start2);
		Node *ptr;
		ptr = start2;
		while(ptr->next!=NULL)
		{
			ptr = ptr->next;
```

```
                }
                Node *newnode = new Node();
                newnode->data = start->data;
                newnode->next = NULL;
                ptr->next = newnode;
                return(start2);
        }
}

int main()
{
        string stri, rev="";
        cout<<"Enter the string : ";
        cin>>stri;
        int i;
        Node *start;
        Node *ptr;
        start = new Node();
        ptr = start;
        for(i=0;i<stri.length();i++)
        {
                Node *newnode = new Node();
                newnode->data = stri[i];
                newnode->next = NULL;
                ptr->next = newnode;
                ptr = newnode;
```

```
        }
        Node *start2;
        start2 = reverse(start, start2);
        Node* ptr2;
        ptr2 = start2;
        while(ptr2!=NULL)
        {
                rev += ptr2->data;
                ptr2 = ptr2->next;
        }
        cout<<"The reversed string is : "<<rev;
}
*/
```




```
/*
//Q2
```

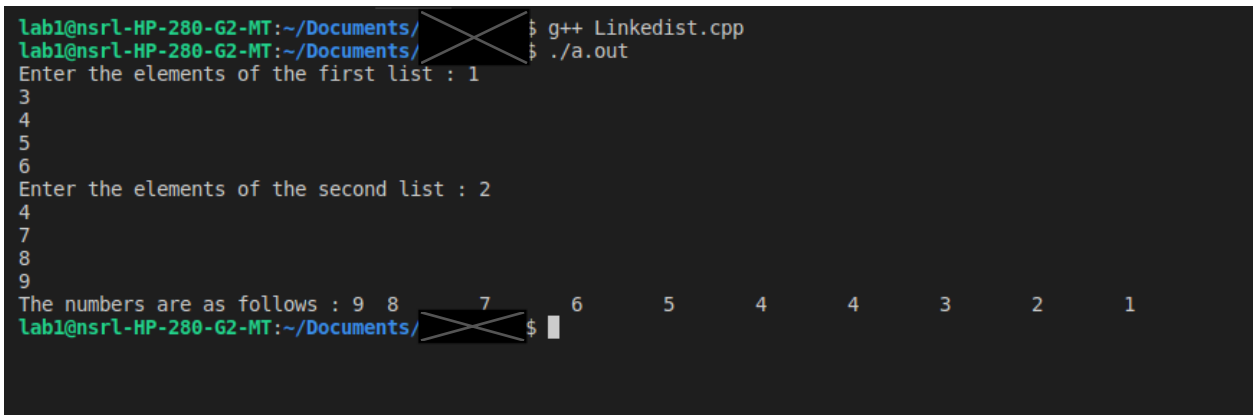
```
#include<iostream>
using namespace std;

class Node
{
	public :
		int data;
		Node *next;
};

Node* reverse(Node *start, Node *start2)
{
	if(start->next == NULL)
	{
		return(start);
	}
	else
	{
		start2 = reverse(start->next, start2);
		Node *ptr;
		ptr = start2;
		while(ptr->next!=NULL)
		{
			ptr = ptr->next;
		}
		Node *newnode = new Node();
```

```
                newnode->data = start->data;
                newnode->next = NULL;
                ptr->next = newnode;
                return(start2);
        }
}

int main()
{
        cout<<"Enter the elements of the first list : ";
        Node *start1 = new Node();
        Node *ptr = start1;
        int i;
        for(i=0;i<5;i++)
        {
                Node *newnode = new Node();
                cin>>newnode->data;
                newnode->next = NULL;
                ptr->next = newnode;
                ptr = newnode;
        }
        Node *start1rev ;
        start1rev = reverse(start1, start1rev);

        cout<<"Enter the elements of the second list : ";
        Node *start2 = new Node();
```

```
ptr = start2;
for(i=0;i<5;i++)
{
        Node *newnode = new Node();
        cin>>newnode->data;
        newnode->next = NULL;
        ptr->next = newnode;
        ptr = newnode;
}
Node *start2rev ;
start2rev = reverse(start2, start2rev);

Node *start = new Node();
Node* ptr1, *ptr2;
ptr1 = start1rev;
ptr2 = start2rev;
ptr = start;
while(true)
{
        if(ptr1==NULL)
        {
                while(ptr1!=NULL)
                {
                        Node *newnode = new Node();
                        newnode->data = ptr1->data;
                        ptr->next = newnode;
```

```
                newnode->next = NULL;
                ptr = newnode;
                ptr1 = ptr1->next;
        }
        break;
}
else if(ptr2==NULL)
{
        while(ptr2!=NULL)
        {
                Node *newnode = new Node();
                newnode->data = ptr2->data;
                ptr->next = newnode;
                newnode->next = NULL;
                ptr = newnode;
                ptr2 = ptr2->next;
        }
        break;
}
else
{
        Node *newnode = new Node();
        if(ptr1->data>ptr2->data)
        {
                newnode->data = ptr1->data;
                ptr1 = ptr1->next;
```

```
                        }
                        else
                        {
                                newnode->data = ptr2->data;
                                ptr2 = ptr2->next;
                        }
                        ptr->next = newnode;
                        newnode->next = NULL;
                        ptr = newnode;
                }
        }
        ptr = start->next;
        cout<<"The numbers are as follows : ";
        while(ptr->next!=NULL)
        {
                cout<<ptr->data<<'\t';
                ptr = ptr->next;
        }
        cout<<endl;
}
*/
```



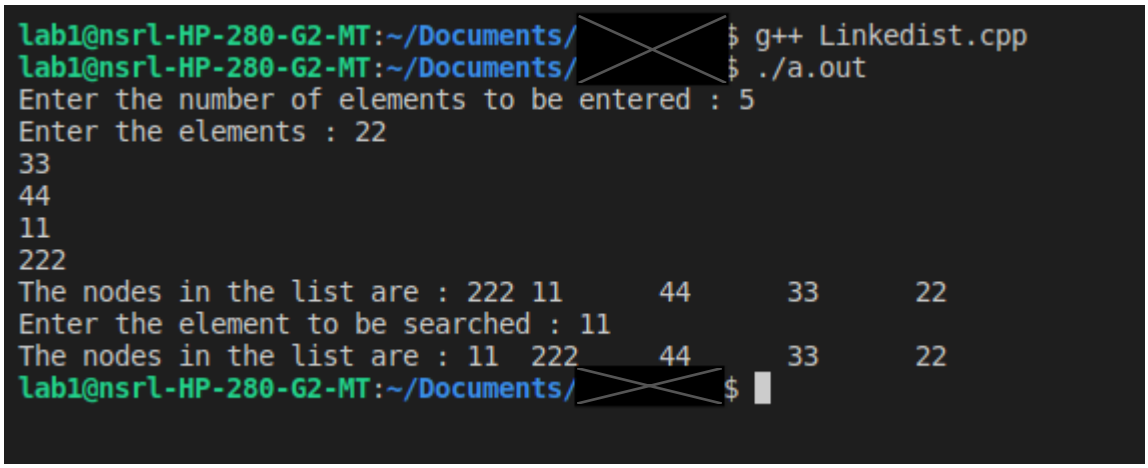
```
//Q3

#include<iostream>
using namespace std;

class Node
{
   //-----------------------------------------Creation of Node----------------------------------------
   public :
      int data;
      Node *next;
};

class Linkedlist
{
   public :
      Node *start, *ptr;

      Linkedlist()
      {
```

```
    start = NULL;
}

void insertFront(int n)
{
    //-----------------------------------------Insertion in Front-----------------------------------------
    Node *newnode = new Node();
    newnode->data = n;
    newnode->next = NULL;
    if(start == NULL)
    {
        start = newnode;
    }
    else
    {
        newnode->next = start;
        start = newnode;
    }
}

void search(int n)
{
    ptr = start;
    bool flag = false;
    if(start->data == n)
    {
```

```
        cout<<"Element fount in the front of the list."<<endl;
    }
    else
    {
        while(ptr->next!=NULL)
        {
            if(ptr->next->data==n)
            {
                flag = true;
                Node *node;
                node = ptr->next;
                ptr->next = node->next;
                node->next = NULL;
                insertFront(n);
                break;
            }
            ptr = ptr->next;
        }
        if(flag==false)
        {
            cout<<"The element is not found."<<endl;
        }
    }
}

void printList()
```

```
    {
        //-----------------------------------------Printing Nodes-----------------------------------------
        cout<<"The nodes in the list are : ";
        ptr = start;
        while(ptr!=NULL)
        {
            cout<<ptr->data<<'\t';
            ptr = ptr->next;
        }
        cout<<endl;
    }
};

int main()
{
    int num;
    cout<<"Enter the number of elements to be entered : ";
    cin>>num;
    int i, n;
    Linkedlist list;
    cout<<"Enter the elements : ";
    for(i=0;i<num;i++)
    {
        cin>>n;
        list.insertFront(n);
    }
```

```
    list.printList();
    cout<<"Enter the element to be searched : ";
    cin>>n;
    list.search(n);
    list.printList();
}
```



//Q4)

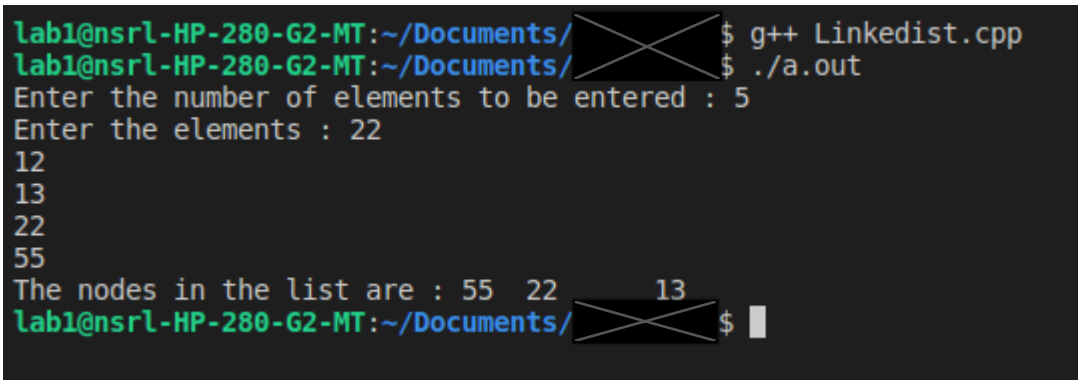
```
#include<iostream>
using namespace std;

class Node
{
    //----------------------------------------Creation of Node----------------------------------------
    public :
        int data;
        Node *next;
```

```
};

class Linkedlist
{
   public :
      Node *start, *ptr;

      Linkedlist()
      {
         start = NULL;
      }

      void insertFront(int n)
      {
         //-----------------------------------------Insertion in Front-----------------------------------------
         Node *newnode = new Node();
         newnode->data = n;
         newnode->next = NULL;
         if(start == NULL)
         {
            start = newnode;
         }
         else
         {
            newnode->next = start;
            start = newnode;
```

```
    }
}

void printList()
{
    //-----------------------------------------Printing Nodes-----------------------------------------
    cout<<"The nodes in the list are : ";
    ptr = start;
    while(ptr!=NULL)
    {
        cout<<ptr->data<<'\t';
        ptr = ptr->next;
    }
    cout<<endl;
}

void deleteList()
{
    bool flag = true;
    Node *ptr;
    while(flag)
    {
        ptr = start;
        if(ptr->next == NULL)
        {
            if(ptr->data %2 ==0)
```

```
            {
               start = new Node();
            }
            break;
         }
         while(ptr->next->next!=NULL)
         {
            ptr = ptr->next;
         }
         if(ptr->next->data % 2 ==0)
         {
            ptr->next = NULL;
         }
         else
         {
            flag = false;
         }

      }
   }
};

int main()
{
   int num;
   cout<<"Enter the number of elements to be entered : ";
```

```
    cin>>num;
    int i, n;
    Linkedlist list;
    cout<<"Enter the elements : ";
    for(i=0;i<num;i++)
    {
        cin>>n;
        list.insertFront(n);
    }
    list.deleteList();
    list.printList();
}
```




```
//Q5)
#include<iostream>
using namespace std;

class Node
{
```

```
   //-------------------------------------------Creation of Node-------------------------------------------
   public :
      char data;
      Node *next;
};

class Linkedlist
{
   public :
      Node *start, *ptr;

      Linkedlist()
      {
         start = NULL;
      }

      void insertFront(char ch)
      {
         //-------------------------------------------Insertion in Front-------------------------------------------
         Node *newnode = new Node();
         newnode->data = ch;
         newnode->next = NULL;
         if(start == NULL)
         {
            start = newnode;
         }
```

```
        else
        {
            newnode->next = start;
            start = newnode;
        }
    }

    char popFront()
    {
        char ch = start->data;
        if(start->next == NULL)
        {
            start = NULL;
        }
        else
        {
            ptr = start->next;
            start->next = NULL;
            start = ptr;
        }
        return(ch);
    }
};

int main()
{
```

```
    string s1, s2, s3;
    cout<<"Enter string 1 : ";
    cin>>s1;
    cout<<"Enter string 2 : ";
    cin>>s2;
    int i;
    Linkedlist list;
    for(i=0;i<s1.length();i++)
    {
       list.insertFront(s1[i]);
    }
    for(i=0;i<s2.length();i++)
    {
       list.insertFront(s2[i]);
    }
    s3 = "";
    for(i=0;i<s1.length()+s2.length();i++)
    {
       s3 += list.popFront();
    }
    cout<<"The final answer is : "<<s3;
    cout<<endl;
}
```

```
lab1@nsrl-HP-280-G2-MT:~/Documents/[redacted]$ g++ Linkedist.cpp
lab1@nsrl-HP-280-G2-MT:~/Documents/[redacted]$ ./a.out
Enter string 1 : abc
Enter string 2 : def
The final answer is : fedcba
lab1@nsrl-HP-280-G2-MT:~/Documents/[redacted]$
```