Java Assignment -7

COLLECTIONS

/\* QUESTION 1:

Write a program to perform string operations using ArrayList. Write functions for the

following

Append - add at end

Insert – add at particular index

Find the index of a particular element (Search)

Display the list

List all string starts with given letter

List of all string contains the Substring

Sort the elements in ArrayList

Remove a particular element

Replace one string with another string in ArrayList

Remove duplicate elements

\*/

import java.util.\*;

import java.util.Collections;

import java.util.LinkedHashSet;

public class ArrayListString{

public void append(String str,ArrayList<String> al){

al.add(str);

}

public void appendWithIndex(String str,int indx,ArrayList<String> al){

al.add(indx,str);

}

public void display(ArrayList<String> al){

System.out.println("Elements of the List : "+al);

}

public int findIndex(String str,ArrayList<String> al){

return al.indexOf(str);

}

public void startWithLtr(char ch,ArrayList<String> al){Iterator<String> itr = al.iterator();

while(itr.hasNext()){

String ele=itr.next();

char chr=ele.charAt(0);

if(ch==chr)

System.out.println(itr.next());

}

}

public void sort(ArrayList<String> al){

Collections.sort(al);

System.out.println("Sorted List: "+al);

}

public int removeString(String str,ArrayList<String> al){

Iterator<String> itr = al.iterator();

while(itr.hasNext()){

String element = itr.next();

if(element.equals(str)){

while(al.indexOf(str)!=-1)

al.remove(al.indexOf(str));

System.out.println("Element Removed!");

return 1;

}

}

return -1;

}

public int replaceString(String str1,String str2,ArrayList<String> al){

int indx=al.indexOf(str1);

if(indx!=-1){

al.set(indx,str2);

return 1;

}

else

return -1;

}

public void removeDuplicates(ArrayList<String> al){

LinkedHashSet<String> hashSet = new LinkedHashSet<>(al);

ArrayList<String> listWithoutDuplicates = new ArrayList<>(hashSet);

System.out.println("List after Removing Duplicates :

"+listWithoutDuplicates);

}

public void findSubstring(String str,ArrayList<String> al){

int flag=0;Iterator<String> itr = al.iterator();

System.out.println("Elements : ");

while(itr.hasNext()) {

String element = itr.next();

if(element.contains(str)){

System.out.print(" "+element);

flag=1;

}

}

if(flag==0)

System.out.println("Substring not found");

}

public static void main(String args[]){

int wish,opt,indx,itr=0;

boolean found;

String str,ch;

char chr;

Scanner obj=new Scanner(System.in);

ArrayListString als=new ArrayListString();

ArrayList<String> al=new ArrayList<String>();

System.out.println("\n\tString Operations using ArrayList\n");

do{

System.out.println("\nEnter the Option: \n\t1 to Append - Add at End\

n\t2 to Insert – Add at Particular Index\n\t3 to Find the Index of a Particular Element\n\

t4 to Display the List\n\t5 to List all Strings starting with given Letter\n\t6 to List all

String Containing a Substring\n\t7 to Sort the Elements in ArrayList\n\t8 to Remove a

Particular Element\n\t9 to Replace one string with another string in ArrayList\n\t10 to

Remove the Duplicate Elements: ");

opt=obj.nextInt();

switch(opt){

case 1 :

System.out.println("\nEnter String to Add: ");

obj.nextLine();

str=obj.nextLine();itr++;

als.append(str,al);

break;

case 2:

System.out.println("Enter Index to Insert (From 0 to

"+itr+"): ");

indx=obj.nextInt();System.out.println("\nEnter String to Add : ");

obj.nextLine();

str=obj.nextLine();

als.appendWithIndex(str,indx,al);

break;

case 3:

System.out.println("\nEnter String to Find: ");

obj.nextLine();

str=obj.nextLine();

indx=als.findIndex(str,al);

if(indx==-1)

System.out.println("Given String is Not Present in

the ArrayList.");

else

System.out.println("Index: "+indx);

break;

case 4:

als.display(al);

break;

case 5:

System.out.println("Enter a Character: ");

chr=obj.next().charAt(0);

als.startWithLtr(chr,al);

break;

case 6:

System.out.println("Enter the Sub-String to Find: ");

obj.nextLine();

ch=obj.nextLine();

als.findSubstring(ch,al);

break;

case 7:

als.sort(al);

break;

case 8:

System.out.println("\nEnter String to Find: ");

obj.nextLine();

str=obj.nextLine();

indx=als.removeString(str,al);

if(indx==-1)

System.out.println("Given String is not present in

the ArrayList");

break;case 9:

System.out.println("\nEnter String to Find: ");

obj.nextLine();

str=obj.nextLine();

System.out.println("\nEnter String to Replace with: ");

ch=obj.nextLine();

indx=als.replaceString(str,ch,al);

if(indx==-1)

System.out.println("Given String is not present in

the ArrayList");

else

System.out.println("Element Replaced

successfully!");

break;

case 10:

als.removeDuplicates(al);

break;

default:

System.out.println("\nInvalid Input");

}

System.out.println("\nDo you wish to continue?(1-Yes/2-No): ");

wish=obj.nextInt();

}while(wish!=2);

}

}

/\*

OUTPUT:

vishnu@vishnu-G7-7588:~/Desktop/JAVA/EX7$ javac ArrayListString.java

vishnu@vishnu-G7-7588:~/Desktop/JAVA/EX7$ java ArrayListString

String Operations using ArrayList

Enter the Option:

1 to Append - Add at End

2 to Insert – Add at Particular Index

3 to Find the Index of a Particular Element

4 to Display the List

5 to List all Strings starting with given Letter

6 to List all String Containing a Substring7 to Sort the Elements in ArrayList

8 to Remove a Particular Element

9 to Replace one string with another string in ArrayList

10 to Remove the Duplicate Elements:

1

Enter String to Add:

Hi

Do you wish to continue?(1-Yes/2-No):

1

Enter the Option:

1 to Append - Add at End

2 to Insert – Add at Particular Index

3 to Find the Index of a Particular Element

4 to Display the List

5 to List all Strings starting with given Letter

6 to List all String Containing a Substring

7 to Sort the Elements in ArrayList

8 to Remove a Particular Element

9 to Replace one string with another string in ArrayList

10 to Remove the Duplicate Elements:

1

Enter String to Add:

Hey

Do you wish to continue?(1-Yes/2-No):

1

Enter the Option:

1 to Append - Add at End

2 to Insert – Add at Particular Index

3 to Find the Index of a Particular Element

4 to Display the List

5 to List all Strings starting with given Letter

6 to List all String Containing a Substring

7 to Sort the Elements in ArrayList

8 to Remove a Particular Element

9 to Replace one string with another string in ArrayList

10 to Remove the Duplicate Elements:1

Enter String to Add:

Vikram

Do you wish to continue?(1-Yes/2-No):

1

Enter the Option:

1 to Append - Add at End

2 to Insert – Add at Particular Index

3 to Find the Index of a Particular Element

4 to Display the List

5 to List all Strings starting with given Letter

6 to List all String Containing a Substring

7 to Sort the Elements in ArrayList

8 to Remove a Particular Element

9 to Replace one string with another string in ArrayList

10 to Remove the Duplicate Elements:

1

Enter String to Add:

Vish

Do you wish to continue?(1-Yes/2-No):

1

Enter the Option:

1 to Append - Add at End

2 to Insert – Add at Particular Index

3 to Find the Index of a Particular Element

4 to Display the List

5 to List all Strings starting with given Letter

6 to List all String Containing a Substring

7 to Sort the Elements in ArrayList

8 to Remove a Particular Element

9 to Replace one string with another string in ArrayList

10 to Remove the Duplicate Elements:

1

Enter String to Add:

JavaDo you wish to continue?(1-Yes/2-No):

1

Enter the Option:

1 to Append - Add at End

2 to Insert – Add at Particular Index

3 to Find the Index of a Particular Element

4 to Display the List

5 to List all Strings starting with given Letter

6 to List all String Containing a Substring

7 to Sort the Elements in ArrayList

8 to Remove a Particular Element

9 to Replace one string with another string in ArrayList

10 to Remove the Duplicate Elements:

1

Enter String to Add:

JavaProg

Do you wish to continue?(1-Yes/2-No):

1

Enter the Option:

1 to Append - Add at End

2 to Insert – Add at Particular Index

3 to Find the Index of a Particular Element

4 to Display the List

5 to List all Strings starting with given Letter

6 to List all String Containing a Substring

7 to Sort the Elements in ArrayList

8 to Remove a Particular Element

9 to Replace one string with another string in ArrayList

10 to Remove the Duplicate Elements:

2

Enter Index to Insert (From 0 to 6):

3

Enter String to Add :

Machine

Do you wish to continue?(1-Yes/2-No):1

Enter the Option:

1 to Append - Add at End

2 to Insert – Add at Particular Index

3 to Find the Index of a Particular Element

4 to Display the List

5 to List all Strings starting with given Letter

6 to List all String Containing a Substring

7 to Sort the Elements in ArrayList

8 to Remove a Particular Element

9 to Replace one string with another string in ArrayList

10 to Remove the Duplicate Elements:

3

Enter String to Find:

Machine

Index: 3

Do you wish to continue?(1-Yes/2-No):

1

Enter the Option:

1 to Append - Add at End

2 to Insert – Add at Particular Index

3 to Find the Index of a Particular Element

4 to Display the List

5 to List all Strings starting with given Letter

6 to List all String Containing a Substring

7 to Sort the Elements in ArrayList

8 to Remove a Particular Element

9 to Replace one string with another string in ArrayList

10 to Remove the Duplicate Elements:

4

Elements of the List : [Hi, Hey, Vikram, Machine, Vish, Java, JavaProg]

Do you wish to continue?(1-Yes/2-No):

1

Enter the Option:

1 to Append - Add at End

2 to Insert – Add at Particular Index3 to Find the Index of a Particular Element

4 to Display the List

5 to List all Strings starting with given Letter

6 to List all String Containing a Substring

7 to Sort the Elements in ArrayList

8 to Remove a Particular Element

9 to Replace one string with another string in ArrayList

10 to Remove the Duplicate Elements:

5

Enter a Character:

H

Hey

Do you wish to continue?(1-Yes/2-No):

1

Enter the Option:

1 to Append - Add at End

2 to Insert – Add at Particular Index

3 to Find the Index of a Particular Element

4 to Display the List

5 to List all Strings starting with given Letter

6 to List all String Containing a Substring

7 to Sort the Elements in ArrayList

8 to Remove a Particular Element

9 to Replace one string with another string in ArrayList

10 to Remove the Duplicate Elements:

6

Enter the Sub-String to Find:

Java

Elements :

Java JavaProg

Do you wish to continue?(1-Yes/2-No):

1

Enter the Option:

1 to Append - Add at End

2 to Insert – Add at Particular Index

3 to Find the Index of a Particular Element

4 to Display the List

5 to List all Strings starting with given Letter

6 to List all String Containing a Substring7 to Sort the Elements in ArrayList

8 to Remove a Particular Element

9 to Replace one string with another string in ArrayList

10 to Remove the Duplicate Elements:

7

Sorted List: [Hey, Hi, Java, JavaProg, Machine, Vikram, Vish]

Do you wish to continue?(1-Yes/2-No):

1

Enter the Option:

1 to Append - Add at End

2 to Insert – Add at Particular Index

3 to Find the Index of a Particular Element

4 to Display the List

5 to List all Strings starting with given Letter

6 to List all String Containing a Substring

7 to Sort the Elements in ArrayList

8 to Remove a Particular Element

9 to Replace one string with another string in ArrayList

10 to Remove the Duplicate Elements:

8

Enter String to Find:

Hey

Element Removed!

Do you wish to continue?(1-Yes/2-No):

1

Enter the Option:

1 to Append - Add at End

2 to Insert – Add at Particular Index

3 to Find the Index of a Particular Element

4 to Display the List

5 to List all Strings starting with given Letter

6 to List all String Containing a Substring

7 to Sort the Elements in ArrayList

8 to Remove a Particular Element

9 to Replace one string with another string in ArrayList

10 to Remove the Duplicate Elements:

4Elements of the List : [Hi, Java, JavaProg, Machine, Vikram, Vish]

Do you wish to continue?(1-Yes/2-No):

1

Enter the Option:

1 to Append - Add at End

2 to Insert – Add at Particular Index

3 to Find the Index of a Particular Element

4 to Display the List

5 to List all Strings starting with given Letter

6 to List all String Containing a Substring

7 to Sort the Elements in ArrayList

8 to Remove a Particular Element

9 to Replace one string with another string in ArrayList

10 to Remove the Duplicate Elements:

9

Enter String to Find:

Java

Enter String to Replace with:

C++

Element Replaced successfully!

Do you wish to continue?(1-Yes/2-No):

1

Enter the Option:

1 to Append - Add at End

2 to Insert – Add at Particular Index

3 to Find the Index of a Particular Element

4 to Display the List

5 to List all Strings starting with given Letter

6 to List all String Containing a Substring

7 to Sort the Elements in ArrayList

8 to Remove a Particular Element

9 to Replace one string with another string in ArrayList

10 to Remove the Duplicate Elements:

4

Elements of the List : [Hi, C++, JavaProg, Machine, Vikram, Vish]Do you wish to continue?(1-Yes/2-No):

1

Enter the Option:

1 to Append - Add at End

2 to Insert – Add at Particular Index

3 to Find the Index of a Particular Element

4 to Display the List

5 to List all Strings starting with given Letter

6 to List all String Containing a Substring

7 to Sort the Elements in ArrayList

8 to Remove a Particular Element

9 to Replace one string with another string in ArrayList

10 to Remove the Duplicate Elements:

1

Enter String to Add:

Hi

Do you wish to continue?(1-Yes/2-No):

1

Enter the Option:

1 to Append - Add at End

2 to Insert – Add at Particular Index

3 to Find the Index of a Particular Element

4 to Display the List

5 to List all Strings starting with given Letter

6 to List all String Containing a Substring

7 to Sort the Elements in ArrayList

8 to Remove a Particular Element

9 to Replace one string with another string in ArrayList

10 to Remove the Duplicate Elements:

4

Elements of the List : [Hi, C++, JavaProg, Machine, Vikram, Vish, Hi]

Do you wish to continue?(1-Yes/2-No):

1

Enter the Option:

1 to Append - Add at End

2 to Insert – Add at Particular Index3 to Find the Index of a Particular Element

4 to Display the List

5 to List all Strings starting with given Letter

6 to List all String Containing a Substring

7 to Sort the Elements in ArrayList

8 to Remove a Particular Element

9 to Replace one string with another string in ArrayList

10 to Remove the Duplicate Elements:

10

List after Removing Duplicates : [Hi, C++, JavaProg, Machine, Vikram, Vish]

Do you wish to continue?(1-Yes/2-No):

2

\*/

/\* QUESTION 2:

Write a program to get two integer ArrayList. Perform the following operations

Merge the two lists

Find Union of two lists

Find Intersection of two lists

Compare two lists

\*/

import java.util.\*;

import java.util.Collections;

import java.util.HashSet;

public class ArrayListInteger{

public static void main(String args[]){

int opt,wish,num;

boolean equal;

System.out.println("\n\tInteger Manipulation using ArrayList \n");

Scanner obj=new Scanner(System.in);

ArrayListInteger ali=new ArrayListInteger();

ArrayList<Integer> al1=new ArrayList<Integer>();

ArrayList<Integer> al2=new ArrayList<Integer>();do{

System.out.println("Enter An Option :\n\t1.Enter Elements to Lists\n\

t2.Merge the 2 Lists\n\t3.Find Union of the two Lists\n\t4.Find Intersection of the two

Lists\n\t5.Compare the two Lists\n\tEnter Your Choice: ");

opt=obj.nextInt();

switch(opt){

case 1:

System.out.println("\nEnter Integer to Add : ");

System.out.println("\nList 1 : ");

num=obj.nextInt();

al1.add(num);

System.out.println("\nList 2 : ");

num=obj.nextInt();

al2.add(num);

break;

case 2:

al1.addAll(al2);

System.out.println("List After Merging : "+al1);

break;

case 3:

Set<Integer> set = new HashSet<Integer>();

set.addAll(al1);

set.addAll(al2);

System.out.println("Union of the two Lists : "+set);

break;

case 4:

List<Integer> list = new ArrayList<Integer>();

for(Integer t : al1){

if(al2.contains(t))

list.add(t);

}

System.out.println("Intersection of the two Lists : "+list);

break;

case 5:

equal=al1.equals(al2);

if(equal==true)

System.out.println("The two Lists are Equal");

else

System.out.println("The two Lists are Not Equal");

break;

default:

System.out.println("\n\tInvalid Input.");}

System.out.println("\nDo You wish to continue ?(1-Yes/2-No): ");

wish=obj.nextInt();

}while(wish!=2);

}

}

/\*

OUTPUT:

vishnu@vishnu-G7-7588:~/Desktop/JAVA/EX7$ javac ArrayListInteger.java

vishnu@vishnu-G7-7588:~/Desktop/JAVA/EX7$ java ArrayListInteger

Integer Manipulation using ArrayList

Enter An Option :

1.Enter Elements to Lists

2.Merge the 2 Lists

3.Find Union of the two Lists

4.Find Intersection of the two Lists

5.Compare the two Lists

Enter Your Choice:

1

Enter Integer to Add :

List 1 :

1

List 2 :

1

Do You wish to continue ?(1-Yes/2-No):

1

Enter An Option :

1.Enter Elements to Lists

2.Merge the 2 Lists

3.Find Union of the two Lists

4.Find Intersection of the two Lists

5.Compare the two Lists

Enter Your Choice:1

Enter Integer to Add :

List 1 :

2

List 2 :

3

Do You wish to continue ?(1-Yes/2-No):

1

Enter An Option :

1.Enter Elements to Lists

2.Merge the 2 Lists

3.Find Union of the two Lists

4.Find Intersection of the two Lists

5.Compare the two Lists

Enter Your Choice:

1

Enter Integer to Add :

List 1 :

34

List 2 :

35

Do You wish to continue ?(1-Yes/2-No):

1

Enter An Option :

1.Enter Elements to Lists

2.Merge the 2 Lists

3.Find Union of the two Lists

4.Find Intersection of the two Lists

5.Compare the two Lists

Enter Your Choice:

1

Enter Integer to Add :List 1 :

32

List 2 :

32

Do You wish to continue ?(1-Yes/2-No):

1

Enter An Option :

1.Enter Elements to Lists

2.Merge the 2 Lists

3.Find Union of the two Lists

4.Find Intersection of the two Lists

5.Compare the two Lists

Enter Your Choice:

3

Union of the two Lists : [32, 1, 2, 34, 3, 35]

Do You wish to continue ?(1-Yes/2-No):

1

Enter An Option :

1.Enter Elements to Lists

2.Merge the 2 Lists

3.Find Union of the two Lists

4.Find Intersection of the two Lists

5.Compare the two Lists

Enter Your Choice:

4

Intersection of the two Lists : [1, 32]

Do You wish to continue ?(1-Yes/2-No):

1

Enter An Option :

1.Enter Elements to Lists

2.Merge the 2 Lists

3.Find Union of the two Lists

4.Find Intersection of the two Lists

5.Compare the two Lists

Enter Your Choice:

5

The two Lists are Not EqualDo You wish to continue ?(1-Yes/2-No):

1

Enter An Option :

1.Enter Elements to Lists

2.Merge the 2 Lists

3.Find Union of the two Lists

4.Find Intersection of the two Lists

5.Compare the two Lists

Enter Your Choice:

2

List After Merging : [1, 2, 34, 32, 1, 3, 35, 32]

Do You wish to continue ?(1-Yes/2-No):

2

\*/

/\* QUESTION 3:

Using Collection framework, create a doubly linked list of integers and perform the

following operations.

a. Insert element on both sides

b. Delete element on both sides

c. Insert an element at a particular position

d. Delete a particular element

e. Search for a particular element

f.Display list in forward order and backward order

g.Sort the elements in LinkedList

h. Replace one element in the list with another list

i.Remove duplicate elements

\*/

import java.util.\*;

public class DLL{

public static void main(String args[]){

LinkedList<Integer> list = new LinkedList<>();

Iterator it;

Iterator looper;

Scanner s=new Scanner(System.in);

int x,y,i,count;int opt=1;

while(opt!=0){

System.out.println("\n\t\tLINKED LIST MANIPULATION\n");

System.out.println("Enter An Option\n\t1 to Insert At Front\n\t2 to

Insert At End\n\t3 to Insert Before Element\n\t4 to Delete An Element\n\t5 to Search An

Element\n\t6 to Display the List in Forward\n\t7 to Display the List in Reverse\n\t8 to

Sort the List\n\t9 to Replace An Element with Another\n\t10 to Remove Duplicates\n\t0

to Exit\n\tYour Choice: ");

opt=s.nextInt();

switch(opt){

case 1:

System.out.println("Enter an Element: ");

x=s.nextInt();

list.addFirst(x);

break;

case 2:

System.out.println("Enter an Element: ");

x=s.nextInt();

list.add(x);

break;

case 3:

System.out.println("Enter an Element: ");

x=s.nextInt();

System.out.println("Enter Element Before Which to

Insert "+x);

y=s.nextInt();

i=list.indexOf(y);

if(i==-1)

System.out.println(y+" was not found in the

Linked List.\n Element "+x+" was not inserted.");

else

list.add(i,x);

break;

case 4:

System.out.println("Enter an Element: ");

x=s.nextInt();

while(list.indexOf(x)!=-1){

y=list.indexOf(x);

list.remove(y);

}

System.out.println("Element "+x+" was succesfully

removed from the Linked List.");break;

case 5:

System.out.println("Enter an Element: ");

x=s.nextInt();

i=list.indexOf(x);

if(i==-1)

System.out.println("Element "+x+" was not found

in the Linked List.");

else

System.out.println("Element "+x+" was found at

Index "+i);

break;

case 6:

it = list.listIterator();

System.out.println("Elements in Forward Order: ");

while (it.hasNext())

System.out.print(it.next()+ " ");

break;

case 7:

it = list.descendingIterator();

System.out.println("Elements in Backward Order: ");

while (it.hasNext())

System.out.print(it.next()+ " ");

break;

case 8:

Collections.sort(list);

System.out.println("Sorted List: \n");

it = list.listIterator();

System.out.println("Elements in Forward Order: ");

while (it.hasNext())

System.out.print(it.next()+ " ");

break;

case 9:

System.out.println("Enter Replacing Element: ");

x=s.nextInt();

System.out.println("Enter Element to be replaced by

"+x);

y=s.nextInt();

while(list.indexOf(y)!=-1){

list.set(list.indexOf(y),x);

}System.out.println("All instances of "+y+" have been

replaced by "+x+".");

break;

case 10:

LinkedList<Integer> temp = new LinkedList<>();

int size=list.size();

for(i=0;i<size;i++){

x=list.get(i);

if(!temp.contains(x))

temp.add(x);

}

list=temp;

System.out.println("Duplicates have been removed!");

break;

case 0:

System.out.println("\t\tThank You!");

break;

default:

System.out.println("\t\tInvalid Option!");

break;

}

}

}

}

/\*

OUTPUT:

vishnu@vishnu-G7-7588:~/Desktop/JAVA/EX7$ javac DLL.java

vishnu@vishnu-G7-7588:~/Desktop/JAVA/EX7$ java DLL

LINKED LIST MANIPULATION

Enter An Option

1 to Insert At Front

2 to Insert At End

3 to Insert Before Element

4 to Delete An Element

5 to Search An Element

6 to Display the List in Forward

7 to Display the List in Reverse

8 to Sort the List

9 to Replace An Element with Another10 to Remove Duplicates

0 to Exit

Your Choice:

1

Enter an Element:

1

LINKED LIST MANIPULATION

Enter An Option

1 to Insert At Front

2 to Insert At End

3 to Insert Before Element

4 to Delete An Element

5 to Search An Element

6 to Display the List in Forward

7 to Display the List in Reverse

8 to Sort the List

9 to Replace An Element with Another

10 to Remove Duplicates

0 to Exit

Your Choice:

1

Enter an Element:

2

LINKED LIST MANIPULATION

Enter An Option

1 to Insert At Front

2 to Insert At End

3 to Insert Before Element

4 to Delete An Element

5 to Search An Element

6 to Display the List in Forward

7 to Display the List in Reverse

8 to Sort the List

9 to Replace An Element with Another

10 to Remove Duplicates

0 to Exit

Your Choice:

1Enter an Element:

3

LINKED LIST MANIPULATION

Enter An Option

1 to Insert At Front

2 to Insert At End

3 to Insert Before Element

4 to Delete An Element

5 to Search An Element

6 to Display the List in Forward

7 to Display the List in Reverse

8 to Sort the List

9 to Replace An Element with Another

10 to Remove Duplicates

0 to Exit

Your Choice:

2

Enter an Element:

92

LINKED LIST MANIPULATION

Enter An Option

1 to Insert At Front

2 to Insert At End

3 to Insert Before Element

4 to Delete An Element

5 to Search An Element

6 to Display the List in Forward

7 to Display the List in Reverse

8 to Sort the List

9 to Replace An Element with Another

10 to Remove Duplicates

0 to Exit

Your Choice:

1

Enter an Element:

293LINKED LIST MANIPULATION

Enter An Option

1 to Insert At Front

2 to Insert At End

3 to Insert Before Element

4 to Delete An Element

5 to Search An Element

6 to Display the List in Forward

7 to Display the List in Reverse

8 to Sort the List

9 to Replace An Element with Another

10 to Remove Duplicates

0 to Exit

Your Choice:

1

Enter an Element:

23

LINKED LIST MANIPULATION

Enter An Option

1 to Insert At Front

2 to Insert At End

3 to Insert Before Element

4 to Delete An Element

5 to Search An Element

6 to Display the List in Forward

7 to Display the List in Reverse

8 to Sort the List

9 to Replace An Element with Another

10 to Remove Duplicates

0 to Exit

Your Choice:

2

Enter an Element:

38

LINKED LIST MANIPULATION

Enter An Option

1 to Insert At Front2 to Insert At End

3 to Insert Before Element

4 to Delete An Element

5 to Search An Element

6 to Display the List in Forward

7 to Display the List in Reverse

8 to Sort the List

9 to Replace An Element with Another

10 to Remove Duplicates

0 to Exit

Your Choice:

6

Elements in Forward Order:

23 293 3 2 1 92 38

LINKED LIST MANIPULATION

Enter An Option

1 to Insert At Front

2 to Insert At End

3 to Insert Before Element

4 to Delete An Element

5 to Search An Element

6 to Display the List in Forward

7 to Display the List in Reverse

8 to Sort the List

9 to Replace An Element with Another

10 to Remove Duplicates

0 to Exit

Your Choice:

3

Enter an Element:

27

Enter Element Before Which to Insert 27

2

LINKED LIST MANIPULATION

Enter An Option

1 to Insert At Front

2 to Insert At End

3 to Insert Before Element

4 to Delete An Element5 to Search An Element

6 to Display the List in Forward

7 to Display the List in Reverse

8 to Sort the List

9 to Replace An Element with Another

10 to Remove Duplicates

0 to Exit

Your Choice:

5

Enter an Element:

27

Element 27 was found at Index 3

LINKED LIST MANIPULATION

Enter An Option

1 to Insert At Front

2 to Insert At End

3 to Insert Before Element

4 to Delete An Element

5 to Search An Element

6 to Display the List in Forward

7 to Display the List in Reverse

8 to Sort the List

9 to Replace An Element with Another

10 to Remove Duplicates

0 to Exit

Your Choice:

4

Enter an Element:

2

Element 2 was succesfully removed from the Linked List.

LINKED LIST MANIPULATION

Enter An Option

1 to Insert At Front

2 to Insert At End

3 to Insert Before Element

4 to Delete An Element

5 to Search An Element

6 to Display the List in Forward7 to Display the List in Reverse

8 to Sort the List

9 to Replace An Element with Another

10 to Remove Duplicates

0 to Exit

Your Choice:

6

Elements in Forward Order:

23 293 3 27 1 92 38

LINKED LIST MANIPULATION

Enter An Option

1 to Insert At Front

2 to Insert At End

3 to Insert Before Element

4 to Delete An Element

5 to Search An Element

6 to Display the List in Forward

7 to Display the List in Reverse

8 to Sort the List

9 to Replace An Element with Another

10 to Remove Duplicates

0 to Exit

Your Choice:

7

Elements in Backward Order:

38 92 1 27 3 293 23

LINKED LIST MANIPULATION

Enter An Option

1 to Insert At Front

2 to Insert At End

3 to Insert Before Element

4 to Delete An Element

5 to Search An Element

6 to Display the List in Forward

7 to Display the List in Reverse

8 to Sort the List

9 to Replace An Element with Another

10 to Remove Duplicates

0 to Exit

Your Choice:1

Enter an Element:

1

LINKED LIST MANIPULATION

Enter An Option

1 to Insert At Front

2 to Insert At End

3 to Insert Before Element

4 to Delete An Element

5 to Search An Element

6 to Display the List in Forward

7 to Display the List in Reverse

8 to Sort the List

9 to Replace An Element with Another

10 to Remove Duplicates

0 to Exit

Your Choice:

1

Enter an Element:

3

LINKED LIST MANIPULATION

Enter An Option

1 to Insert At Front

2 to Insert At End

3 to Insert Before Element

4 to Delete An Element

5 to Search An Element

6 to Display the List in Forward

7 to Display the List in Reverse

8 to Sort the List

9 to Replace An Element with Another

10 to Remove Duplicates

0 to Exit

Your Choice:

8

Sorted List:Elements in Forward Order:

1 1 3 3 23 27 38 92 293

LINKED LIST MANIPULATION

Enter An Option

1 to Insert At Front

2 to Insert At End

3 to Insert Before Element

4 to Delete An Element

5 to Search An Element

6 to Display the List in Forward

7 to Display the List in Reverse

8 to Sort the List

9 to Replace An Element with Another

10 to Remove Duplicates

0 to Exit

Your Choice:

9

Enter Replacing Element:

31

Enter Element to be replaced by 31

23

All instances of 23 have been replaced by 31.

LINKED LIST MANIPULATION

Enter An Option

1 to Insert At Front

2 to Insert At End

3 to Insert Before Element

4 to Delete An Element

5 to Search An Element

6 to Display the List in Forward

7 to Display the List in Reverse

8 to Sort the List

9 to Replace An Element with Another

10 to Remove Duplicates

0 to Exit

Your Choice:

6

Elements in Forward Order:

1 1 3 3 31 27 38 92 293LINKED LIST MANIPULATION

Enter An Option

1 to Insert At Front

2 to Insert At End

3 to Insert Before Element

4 to Delete An Element

5 to Search An Element

6 to Display the List in Forward

7 to Display the List in Reverse

8 to Sort the List

9 to Replace An Element with Another

10 to Remove Duplicates

0 to Exit

Your Choice:

10

Duplicates have been removed!

LINKED LIST MANIPULATION

Enter An Option

1 to Insert At Front

2 to Insert At End

3 to Insert Before Element

4 to Delete An Element

5 to Search An Element

6 to Display the List in Forward

7 to Display the List in Reverse

8 to Sort the List

9 to Replace An Element with Another

10 to Remove Duplicates

0 to Exit

Your Choice:

6

Elements in Forward Order:

1 3 31 27 38 92 293

LINKED LIST MANIPULATION

Enter An Option

1 to Insert At Front

2 to Insert At End

3 to Insert Before Element4 to Delete An Element

5 to Search An Element

6 to Display the List in Forward

7 to Display the List in Reverse

8 to Sort the List

9 to Replace An Element with Another

10 to Remove Duplicates

0 to Exit

Your Choice:

0

Thank You!