

Questions

QName	Action
find leader element in array	Load
Find sadal point	Load
replace 0 to 1	Load
pattern11	Load
pattern16	Load
pattern17	Load
pattern18	Load
pattern7	Load
pattern9	Load
pattern15	Load
Addition of two matrix	Load
basic sort	Load
Multiplication of matrix	Load
pattern5	Load
transpose of matrix	Load
pattern12	Load
Binary Search	Load
Selection sort	Load

```
{
    //StartOfMainMethod
    public static void main(String[] args)
    {
        int irange=9;
        int jrange=5;
        myCode(irange,jrange);
    }
    //EndOfMainMethod
    private static void myCode(int irange,int jrange)
    {
        loopI(0,irange,jrange);
        System.out.println();
    }
    public static void loopI(int i,int irange,int jrange)
    {
        if(i<irange)
        {
            loopj(i,0,jrange);
            System.out.println();
            i++;
            loopI(i,irange,jrange);
        }
    }
    public static void loopj(int i,int j,int jrange)
    {
        if(j<jrange)
        {
            if(j+i<=jrange-1||j-i<=(jrange-1))
                System.out.print("*");
            else
            {
                System.out.print(" ");
            }
            j++;
            loopj(i,j,jrange);
        }
    }
}
```

1	Addition of two matrix	Load
2	pattern7	Load
3	pattern16	Load
4	pattern9	Load
5	transpose of matrix	Load
6	find leader element in array	Load
7	pattern5	Load
8	pattern12	Load
9	pattern17	Load
10	Find sadal point	Load
11	Binary Search	Load
12	replace 0 to 1	Load
13	pattern18	Load
14	remove1stFrom2nd	Load
15	pattern15	Load
16	basic sort	Load
17	Selection sort	Load
18	pattern11	Load
19	Multiplication of matrix	Load
20	reverse Array using third variable	Load

```

{
    //WriteCode Here
    int i=0;
    LoopI(i,irange,jrange);

    public static void LoopI(int i,int irange,int jrange)
    {
        if(i<irange)
        {
            int j=0;
            LoopJ(i,j,jrange);
            System.out.println();
            i++;
            LoopI(i,irange,jrange);
        }
        else
        return;
    }

    public static void LoopJ(int i,int j,int jrange)
    {
        if(j<jrange)
        {
            if(j-i>=0)
            {
                System.out.print("*");
            }
            else
            {
                System.out.print(" ");
            }
            j++;
            LoopJ(i,j,jrange);
        }
        else
        return;
    }
}

```

ideQuestions

QName	Action
array contain specific value	Load
removeASpecifiedCharacterFrom AString	Load
check equality of two array	Load
find leader element in array	Load
SecondMostFrequentChar	Load
find second max	Load
FindFrequencyOfeachCharacter	Load
GivenStringStarts WithTheContentsOfAnotherString	Load
remove duplicate	Load
stringContainsOnlyDigits	Load
printMax EElement of Matrix array	Load
avg of inner element of 2D array	Load
compareAGivenStringToTheSpecifiedcharacterSequence	Load
basic sort	Load
cipher exchanges	Load
interleaving of two array	Load
sortNumbersInAString.	Load
Find sadal point	Load
sumOf TheDigitsPresentInTheGivenString	Load
removeDuplicateEleFromString	Load

```

1 //StartOfMainMethod
2 public static void main(String[] args)
3 {
4     int[][] a=
5     {
6         {6,3,9}
7     }
8     {
9         {9,7,8}
10    }
11    ,
12    {
13        {2,4,5}
14    }
15    ,
16    }
17    ;
18    myCode(a);
19 }
20 //EndOfMainMethod
21 private static void myCode(int[][] a)
22 {
23     int count=0;
24     count=loopi(a,0,count);
25     if(count==0)
26     {
27         System.out.println("no saddle point");
28     }
29     private static int loopi(int[][]a,int i,int count)
30     {
31         if(i<a.length)
32         {
33             count=loopj(a,i,0,count);
34             i++;
35             return loopi(a,i,count);
36         }
37         else
38             return count;
39     }
40     private static int loopj(int[][]a,int i,int j,int count)
41     {
42         if(j<a[i].length)
43         {
44             boolean rowmechota=isrowmechota(a,i,j);
45             boolean colmebada=iscolmebada(a,i,j);
46             if(rowmechota && colmebada)
47             {

```

[HideQuestions](#)

SN	QName	Action
1	array contain specific value	Load
2	removeASpecifiedCharacterFrom AString	Load
3	check equality of two array	Load
4	find leader element in array	Load
5	SecondMostFrequentChar	Load
6	find second max	Load
7	FindFrequencyOfeachCharacter	Load
8	GivenStringStartsWithTheContentsOfAnotherString	Load
9	remove duplicate	Load
10	stringContainsOnlyDigits	Load
11	printMax EElement of Matrix array	Load
12	avg of inner element of 2D array	Load
13	compareAGivenStringToTheSpecifiedcharacterSequence	Load
14	basic sort	Load
15	cipher exchanges	Load
16	interleaving of two array	Load
17	sortNumbersInAString.	Load
18	Find sadal point	Load
19	sumOf TheDigitsPresentInTheGivenString	Load
20	removeDuplicateEleFromString	Load

yourCode

```
1         boolean colmebada=iscolmebada(a,i,j);
2         if(rowmechota && colmebada)
3         {
4             System.out.println(a[i][j]);
5             count++;
6             System.exit(0);
7         }
8         j++;
9     }
10    return count;
11 }
12 private static boolean iscolmebada(int[][]a,int i,int j)
13 {
14     boolean result=true;
15     result =loopcol(a,i,j,0,result);
16     return result;
17 }
18 private static boolean loopcol(int[][]a,int i,int j,int k,boolean result)
19 {
20     if(k<a.length)
21     {
22         if(a[k][j]>a[i][j])
23             result=false;
24         k++;
25     }
26     return result;
27 }
28 private static boolean isrowmechota(int[][]a,int i,int j)
29 {
30     boolean result=true;
31     result=looprow(a,i,j,0,true);
32     return result;
33 }
34 private static boolean looprow(int[][]a,int i, int j, int k,boolean result)
35 {
36     if(k<a[i].length)
37     {
38         if(a[i][k]<a[i][j])
39         {
40             result=false;
41         }
42         k++;
43     }
44     return looprow(a,i,j,k,result);
45 }
46 else
47     return result;}}}
```

AllowYourCode | Delete | Edit this Question | Print | Submit

HideQuestions		
SN	QName	Action
1	basic sort	Load
2	stringContainsOnlyDigits	Load
3	compareAGivenStringToTheSpecifiedcharacterSequence	Load
4	avg of inner element of 2D array	Load
5	sumOf TheDigitsPresentInTheGivenString	Load
6	removeDuplicateEleFromString	Load
7	array contain specific value	Load
8	Find sadal point	Load
9	cipher exchanges	Load
10	SecondMostFrequentChar	Load
11	FindFrequencyOfeachCharacter	Load
12	printMax EElement of Matrix array	Load
13	check equality of two array	Load
14	removeASpecifiedCharacterFrom AString	Load
15	GivenStringStartsWithTheContentsOfAnotherString	Load
16	find second max	Load
17	sortNumbersInAString.	Load
18	find leader element in array	Load
19	interleaving of two array	Load
20	remove duplicate	Load

yourCode

```

1 public class MyClass
2 {
3     static int cnt;
4     static int sum;
5     //StartOfMainMethod
6     public static void main(String[] args)
7     {
8         String s1="a2ba1ca3d";
9         myCode(s1);
10    }
11    //EndOfMainMethod
12    private static void myCode(String s1)
13    {
14        //WriteCode Here
15        loopi(0,s1);
16        if(cnt==0)
17        {
18            System.out.println(0);
19        }
20        else
21        {
22            System.out.println(sum);
23        }
24    }
25    private static void loopi(int i,String s1)
26    {
27        if(i<s1.length())
28        {
29            if(s1.charAt(i)>='0'&&s1.charAt(i)<='9')
30            {
31                int num=s1.charAt(i)-48;
32                sum=sum+num;
33                cnt++;
34            }
35            i++;
36            loopi(i,s1);
37        }
38    }
39 }

```

	return
array contain specific value	Load
removeASpecifiedCharacterFrom AString	Load
check equality of two array	Load
find leader element in array	Load
SecondMostFrequentChar	Load
find second max	Load
FindFrequencyOfeachCharacter	Load
GivenStringStartsWithTheContentsOfAnotherString	Load
remove duplicate	Load
stringContainsOnlyDigits	Load
printMax EElement of Matrix array	Load
avg of inner element of 2D array	Load
compareAGivenStringToTheSpecifiedcharacterSequence	Load
basic sort	Load
cipher exchanges	Load
interleaving of two array	Load
sortNumbersInAString.	Load
Find sadal point	Load
sumOf TheDigitsPresentInTheGivenString	Load
removeDuplicateEleFromString	Load

```

1 public class MyClass
2 {
3     //StartOfMainMethod
4     public static void main(String[] args)
5     {
6         int[] a =
7         {
8             2, 4, 3, 5, 6, -2, 4, 7, 9, 8, 9
9         }
10    ;
11    myCode(a);
12    //EndOfMainMethod
13    private static void myCode(int[] a)
14    {
15        int i=0;
16        int max=Integer.MIN_VALUE;
17        int secmax=0;
18        secmax=loopi(a,i,max,secmax);
19        System.out.println(secmax);
20    }
21    public static int loopi(int[]a,int i,int max,int secmax)
22    {
23        if(i<a.length)
24        {
25            if(a[i]>max)
26            {
27                secmax=max;
28                max=a[i];
29            }
30            else if(a[i]!=max && a[i]>secmax)
31            {
32                secmax=a[i];
33            }
34            i++;
35        }
36        return loopi(a,i,max,secmax);
37    }
38    else
39    return secmax;
40    }
41 }

```

GivenStringStarts With The Contents Of Another String	Load
remove duplicate	Load
string Contains Only Digits	Load
print Max Element of Matrix array	Load
avg of inner element of 2D array	Load
compare A Given String To The Specified character Sequence	Load
basic sort	Load
cipher exchanges	Load
interleaving of two array	Load
sortNumbersInAString.	Load
Find sadal point	Load
sumOf The Digits Present In The Given String	Load
removeDuplicateEle From String	Load

```

1  port java.util.Arrays;
2  import java.util.Iterator;
3  public class MyClass
4  {
5      //StartOfMainMethod
6      public static void main(String[] args)
7      {
8          String s1 = "abbcfedafbcd";
9          myCode(s1);
10     }
11     //EndOfMainMethod
12     private static void myCode(String s1)
13     {
14         String s="";
15         s=loopi(s1,s,0);
16         System.out.print(s);
17     }
18     private static String loopi(String s1,String s,int i)
19     {
20         if(i<s1.length())
21         {
22             int count=0;
23             count=loopj(s1,s,i,i-1,count);
24             if(count==0)
25             {
26                 s=s+s1.charAt(i);
27             }
28             i++;
29             return loopi(s1,s,i);
30         }
31         else
32         {
33             return s;
34         }
35     }
36     private static int loopj(String s1,String s,int i,int j,int count)
37     {
38         if(j>=0)
39         {
40             if(s1.charAt(i)==s1.charAt(j))
41                 count++;
42             j--;
43             return loopj(s1,s,i,j,count);
44         }
45         else
46         {
47             return count;
48         }
49     }
50 }
```

Questions	Action
name	Load
ay contain specific value	Load
moveASpecifiedCharacterFrom AString	Load
check equality of two array	Load
nd leader element in array	Load
secondMostFrequentChar	Load
nd second max	Load
indFrequencyOfeachCharacter	Load
GivenStringStartsWithTheContentsOfAnotherString	Load
remove duplicate	Load
stringContainsOnlyDigits	Load
printMax EElement of Matrix array	Load
avg of inner element of 2D array	Load
compareAGivenStringToTheSpecifiedcharacterSequence	Load
basic sort	Load
cipher exchanges	Load
interleaving of two array	Load
sortNumbersInAString.	Load
Find sadal point	Load
sumOf TheDigitsPresentInTheGivenString	Load
removeDuplicateEleFromString	Load

```

1 public class MyClass
2 {
3     static int cnt;
4     static int sum;
5     //StartOfMainMethod
6     public static void main(String[] args)
7     {
8         String s1="a2ba1ca3d";
9         myCode(s1);
10    }
11    //EndOfMainMethod
12    private static void myCode(String s1)
13    {
14        loopi(0,s1);
15        if(cnt==0)
16        {
17            System.out.println(0);
18        }
19        else
20        {
21            System.out.println(sum);
22        }
23    }
24    private static void loopi(int i, String s1)
25    {
26        if(i<s1.length())
27        {
28            if(s1.charAt(i)>='0' && s1.charAt(i)<='9')
29            {
30                int num=s1.charAt(i)-48;
31                sum=sum+num;
32                cnt++;
33            }
34        }
35    }
36 }

```

array contain specific value	Load
removeASpecifiedCharacterFrom AString	Load
check equality of two array	Load
find leader element in array	Load
SecondMostFrequentChar	Load
find second max	Load
FindFrequencyOfeachCharacter	Load
GivenStringStartsWithTheContentsOfAnotherString	Load
remove duplicate	Load
stringContainsOnlyDigits	Load
printMax EElement of Matrix array	Load
avg of inner element of 2D array	Load
compareAGivenStringToTheSpecifiedcharacterSequence	Load
basic sort	Load
cipher exchanges	Load
interleaving of two array	Load
sortNumbersInAString.	Load
Find sadal point	Load
sumOf TheDigitsPresentInTheGivenString	Load
removeDuplicateEleFromString	Load

```

//startOfMainMethod
public static void main(String[] args)
{
    String s1 = "43521";
    myCode(s1);
}
//EndofMainMethod
private static void myCode(String s1)
{
    char[] ch=s1.toCharArray();
    char temp=0;
    loopJ(ch,0,temp);
    print(ch,0);
}
private static void loopJ(char[] ch,int j,char temp)
{
    if(j<ch.length-1)
    {
        if(ch[j]>ch[j+1])
        {
            temp=ch[j];
            ch[j]=ch[j+1];
            ch[j+1]=temp;
            j=-1;
        }
        j++;
        loopJ(ch,j,temp);
    }
}
private static void print(char[] ch,int i)
{
    if(i<ch.length)
    {
        System.out.print(ch[i]);
        i++;
        print(ch,i);
    }
}

```

SecondMostFrequentChar	Load
find second max	Load
FindFrequencyOfeachCharacter	Load
GivenStringStartsWithTheContentsOfAnotherString	Load
remove duplicate	Load
stringContainsOnlyDigits	Load
printMax Element of Matrix array	Load
avg of inner element of 2D array	Load
compareAGivenStringToTheSpecifiedcharacterSequence	Load
basic sort	Load
cipher exchanges	Load
interleaving of two array	Load
sortNumbersInAString.	Load
Find sadal point	Load
sumOf TheDigitsPresentInTheGivenString	Load
removeDuplicateEleFromString	Load

```

3 //StartOfMainMethod
4 public static void main(String[] args)
5 {
6     char[] a= {'a','b','c','d','e','f','g'};
7     char[] b= {'w','x','y','z'};
8     myCode(a,b);
9 }
10 //EndOfMainMethod
11 private static void myCode(char[] a,char[] b)
12 {
13     char[] c=new char[a.length+b.length];
14     int index=0;
15     loopi(a,b,c,0,index);
16     print(0,c);
17 }
18 private static void print(int i,char[] c)
19 {
20     if(i<c.length)
21     {
22         System.out.print(c[i]+" ");
23         i++;
24         print(i,c);
25     }
26 }
27 private static void loopi(char[] a,char[] b,char[] c,int i,int index)
28 {
29     if(i<c.length)
30     {
31         if(i<a.length)
32         {
33             c[index++]=a[i];
34         }
35         if(i<b.length)
36         {
37             c[index++]=b[i];
38         }
39         i++;
40         loopi(a,b,c,i,index);
41     }
42 }

```

sions

name	Action
My contain specific value	Load
removeASpecifiedCharacterFrom AString	Load
check equality of two array	Load
Find leader element in array	Load
SecondMostFrequentChar	Load
Find second max	Load
FindFrequencyOfeachCharacter	Load
GivenStringStartsWithTheContentsOfAnotherString	Load
remove duplicate	Load
stringContainsOnlyDigits	Load
printMax Element of Matrix array	Load
Avg of inner element of 2D array	Load
CompareAGivenStringToTheSpecifiedcharacterSequence	Load
basic sort	Load
Caesar cipher exchanges	Load
Interleaving of two array	Load
sortNumbersInAString.	Load
Find sadal point	Load

```
yourCode
1 import java.util.Arrays;
2 import java.util.Iterator;
3 public class MyClass
4 {
5     //StartOfMainMethod
6     public static void main(String[] args)
7     {
8         String s1 = "abcdXYZ";
9         myCode(s1);
10    }
11    //EndOfMainMethod
12    private static void myCode(String s1)
13    {
14        s1=s1.toUpperCase();
15        String s2="";
16        s2=loopi(s1,0,s2);
17        System.out.println(s2);
18    }
19    private static String loopi(String s1,int i,String s2)
20    {
21        if(i<s1.length())
22        {
23            char ch=s1.charAt(i);
24            int num=ch;
25            int num2=155-num;
26            char ch1=(char)num2;
27            s2=s2+ch1;
28            i++;
29        }
30    }
31 }
```

ideQuestions

N	QName	Action
1	array contain specific value	Load
2	removeASpecifiedCharacterFrom AString	Load
3	check equality of two array	Load
4	find leader element in array	Load
5	SecondMostFrequentChar	Load
6	find second max	Load
7	FindFrequencyOfeachCharacter	Load
8	GivenStringStartsWithTheContentsOfAnotherString	Load
9	remove duplicate	Load
10	stringContainsOnlyDigits	Load
11	printMax EElement of Matrix array	Load
12	avg of inner element of 2D array	Load
13	compareAGivenStringToTheSpecifiedcharacterSequence	Load
14	basic sort	Load
15	cipher exchanges	Load
16	interleaving of two array	Load
17	sortNumbersInAString.	Load
18	Find sadal point	Load
19	sumOf TheDigitsPresentInTheGivenString	Load
20	removeDuplicateEleFromString	Load

```

3 //StartOfMainMethod
4 public static void main(String[] args)
5 {
6     int[][] a= {{1,2,3,4},
7                  {5,6,7,8 },
8                  {9,4,2,5},
9                  {7,2,4,9}};
10    myCode(a);
11 }
12 //EndOfMainMethod
13 private static void myCode(int[][] a)
14 {
15     int i=0;
16     int[]result=new int[2];
17     result=iloop(a,i,result);
18     System.out.println(result[0]);
19     double avg=result[0]/result[1];
20     System.out.println(avg);
21 }
22 public static int[]iloop(int[][]a,int i,int[] result)
23 {
24     if(i<a.length)
25     {
26         int j=0;
27         result=jloop(a,i,j,result);
28         i++;
29         return iloop(a,i,result);
30     }
31     else
32         return result;
33 }
34 public static int[]jloop(int [][]a,int i,int j,int[]result)
35 {
36     if(j<a[i].length)
37     {
38         if(i!=0 &&j!=0 && i!=a.length-1 &&j!=a[i].length-1)
39         {
40             result[0]=result[0]+a[i][j];
41             result[1]++;
42         }
43         j++;
44     }
45     return jloop(a,i,j,result);
46 }
47 else
48     return result;
49 }
50 }
```

eQuestions

QName	Action
array contain specific value	Load
removeASpecifiedCharacterFrom AString	Load
check equality of two array	Load
find leader element in array	Load
SecondMostFrequentChar	Load
find second max	Load
FindFrequencyOfeachCharacter	Load
GivenStringStartsWithTheContentsOfAnotherString	Load
remove duplicate	Load
stringContainsOnlyDigits	Load
printMax EElement of Matrix array	Load
avg of inner element of 2D array	Load
compareAGivenStringToTheSpecifiedcharacterSequence	Load
basic sort	Load
cipher exchanges	Load
interleaving of two array	Load
sortNumbersInAString.	Load
Find sadal point	Load
sumOf TheDigitsPresentInTheGivenString	Load
removeDuplicateEleFromString	Load

```

1 public class MyClass
2 {
3     //StartOfMainMethod
4     public static void main(String[] args)
5     {
6         int[][] a= {{1,2,3},
7                     {4,23,5},
8                     {5,4,7}};
9         myCode(a);
10    }
11    //EndOfMainMethod
12    private static void myCode(int[][] a)
13    {
14        int max=Integer.MIN_VALUE;
15        max=loopi(a,0,max);
16        System.out.println(max);
17    }
18    private static int loopi(int[][]a,int i, int max)
19    {
20        if(i<a.length)
21        {
22            max=loopj(a,i,0,max);
23            i++;
24            return loopi(a,i,max);
25        }
26        else
27        {
28            return max;
29        }
30    }
31    private static int loopj(int a[][],int i,int j,int max)
32    {
33        if(j<a[i].length)
34        {
35            if(a[i][j]>max)
36            {
37                max=a[i][j];
38            }
39            j++;
40            return loopj(a,i,j,max);
41        }
42        else
43        {
44            return max;
45        }
46    }
47 }
```

[HideQuestions](#)

SN	QName	Action
1	array contain specific value	Load
2	removeASpecifiedCharacterFrom AString	Load
3	check equality of two array	Load
4	find leader element in array	Load
5	SecondMostFrequentChar	Load
6	find second max	Load
7	FindFrequencyOfeachCharacter	Load
8	GivenStringStartsWithTheContentsOfAnotherString	Load
9	remove duplicate	Load
10	stringContainsOnlyDigits	Load
11	printMax EElement of Matrix array	Load
12	avg of inner element of 2D array	Load
13	compareAGivenStringToTheSpecifiedcharacterSequence	Load
14	basic sort	Load
15	cipher exchanges	Load
16	interleaving of two array	Load
17	sortNumbersInAString.	Load
18	Find sadal point	Load
19	sumOf TheDigitsPresentInTheGivenString	Load
20	removeDuplicateEleFromString	Load

yourCode

```
1 public class MyClass
2 {
3     //StartOfMainMethod
4     public static void main(String[] args)
5     {
6         String s1 ="1f23";
7         myCode(s1);
8     }
9     //EndOfMainMethod
10    private static void myCode(String s1)
11    {
12        //write code here
13        int cnt=0;
14        cnt=loopI(s1,0,cnt);
15        if(cnt==0)
16        {
17            System.out.println("contains only digits");
18        }
19        else
20        {
21            System.out.println("not contains only digit");
22        }
23    }
24    private static int loopI(String s1,int i,int cnt)
25    {
26        if(i<s1.length())
27        {
28            char ch=s1.charAt(i);
29            if(ch<'0' || ch>'9')
30            {
31                cnt++;
32            }
33            i++;
34        }
35        return loopI(s1,i,cnt);
36    }
37    else
38    {
39        return cnt;
40    }
41 }
```

HideQuestions

SN	QName	Action
1	array contain specific value	Load
2	removeASpecifiedCharacterFrom AString	Load
3	check equality of two array	Load
4	find leader element in array	Load
5	SecondMostFrequentChar	Load
6	find second max	Load
7	FindFrequencyOfeachCharacter	Load
8	GivenStringStartsWithTheContentsOfAnotherString	Load
9	remove duplicate	Load
10	stringContainsOnlyDigits	Load
11	printMax Element of Matrix array	Load
12	avg of inner element of 2D array	Load
13	compareAGivenStringToTheSpecifiedcharacterSequence	Load
14	basic sort	Load
15	cipher exchanges	Load
16	interleaving of two array	Load
17	sortNumbersInAString.	Load
18	Find saddle point	Load
19	sumOf TheDigitsPresentInTheGivenString	Load
20	removeDuplicateEleFromString	Load

```

2 { //StartOfMainMethod
3     public static void main(String[] args)
4     {
5         int[] a= {1,3,5,6,2,3,4,5,97,1,23,};
6         myCode(a);
7     }
8     //EndOfMainMethod
9     private static void myCode(int[] a)
10    {
11        int dcount=0;
12        dcount=iloop(a,0,dcount);
13        int b[] =new int[a.length-dcount];
14        int bindex=0;
15        loopi(a,0,b,bindex);
16        print(b,0);
17    }
18    private static void print(int []b,int i)
19    {
20        if(i<b.length)
21        {
22            System.out.print(b[i]+" ");
23            i++;
24            print(b,i);
25        }
26    }
27    private static void loopi(int[]a,int i,int[]b,int bindex)
28    {
29        if(i<a.length)
30        {
31            int count=0;
32            count=loopj(a,i,b,0,count);
33            if(count==0)
34            {
35                b[bindex++]=a[i];
36                i++;
37                loopi(a,i,b,bindex);
38            }
39        }
40        private static int loopj(int []a,int i, int[]b,int j,int count)
41        {
42            if(j<b.length)
43            {
44                if(a[i]==b[j])
45                {
46                    count++;
47                    j++;
48                    return loopj(a,i,b,j,count);
49                }
50            }
51            else
52            {
53                return count;
54            }
55        }
56        private static int iloop(int []a,int i,int dcount)
57        {
58            if(i<a.length)
59            {
60                dcount=jloop(a,i,i+1,dcount);
61                i++;
62                return iloop(a,i,dcount);
63            }
64            else
65            {
66                return dcount;
67            }
68        }
69        private static int jloop(int a[],int i,int j,int dcount)
70        {
71            if(j<a.length)
72            {
73                if(a[i]==a[j])
74                {
75                    dcount++;
76                    return dcount;
77                }
78                j++;
79                return jloop(a,i,j,dcount);
80            }
81            else
82            {
83                return dcount;
84            }
85        }
86    }
87 }

```

HideQuestions

SN	QName	Action
1	array contain specific value	Load
2	removeASpecifiedCharacterFrom AString	Load
3	check equality of two array	Load
4	find leader element in array	Load
5	SecondMostFrequentChar	Load
6	find second max	Load
7	FindFrequencyOfeachCharacter	Load
8	GivenStringStartsWithTheContentsOfAnotherString	Load
9	remove duplicate	Load
10	stringContainsOnlyDigits	Load
11	printMax EElement of Matrix array	Load
12	avg of inner element of 2D array	Load
13	compareAGivenStringToTheSpecifiedcharacterSequence	Load
14	basic sort	Load
15	cipher exchanges	Load
16	interleaving of two array	Load
17	sortNumbersInAString.	Load
18	Find sadal point	Load
19	sumOf TheDigitsPresentInTheGivenString	Load
20	removeDuplicateEleFromString	Load

```
yourCode
1 public class MyClass
2 {
3     //StartOfMainMethod
4     public static void main(String[] args)
5     {
6         String s1 ="Priyaabcd";
7         String s2 ="abcd";
8         System.out.println("java method : "+s1.startsWith(s2));
9         myCode(s1,s2);
10    }
11    //EndOfMainMethod
12    private static void myCode(String s1,String s2)
13    {
14        char ch1[]=s1.toCharArray();
15        char ch2[]=s2.toCharArray();
16        if(ch1.length==ch2.length)
17        {
18            int count=0;
19            count=loopi(0,ch1,ch2,count);
20            if(count==0)
21            {
22                System.out.println(true);
23            }
24        }
25        else
26        System.out.println(false);
27    }
28    private static int loopi(int i,char[] ch1,char[] ch2,int count)
29    {
30        if(i<ch2.length)
31        {
32            int num1=ch1[i];
33            int num2=ch2[i];
34            if(num1!=num2)
35            {
36                count++;
37                System.out.println(false);
38            }
39        }
40        i++;
41        count=loopi(i,ch1,ch2,count);
42    }
43    return count;
44 }
```

-hideQuestions

SN	QName	Action
1	array contain specific value	Load
2	removeASpecifiedCharacterFrom AString	Load
3	check equality of two array	Load
4	find leader element in array	Load
5	SecondMostFrequentChar	Load
6	find second max	Load
7	FindFrequencyOfeachCharacter	Load
8	GivenStringStartsWithTheContentsOfAnotherString	Load
9	remove duplicate	Load
10	stringContainsOnlyDigits	Load
11	printMax Element of Matrix array	Load
12	avg of inner element of 2D array	Load
13	compareAGivenStringToTheSpecifiedcharacterSequence	Load
14	basic sort	Load
15	cipher exchanges	Load
16	interleaving of two array	Load
17	sortNumbersInAString.	Load
18	Find sadal point	Load
19	sumOf TheDigitsPresentInTheGivenString	Load
20	removeDuplicateEleFromString	Load

yourCode

```
1 public class MyClass
2 {
3     //StartOfMainMethod
4     public static void main(String[] args)
5     {
6         int[] a= {1,15,13,4,5};
7         myCode(a);
8     }
9     //EndOfMainMethod
10    private static void myCode(int[] a)
11    {
12        int i=0;
13        leaderI(a,0);
14    }
15    private static void leaderI(int []a,int i)
16    {
17        if(i<a.length)
18        {
19            int j=i+1;
20            int count=0;
21            count=leaderJ(a,i,j,count);
22            if(count==0)
23            {
24                System.out.println(a[i]);
25            }
26            i++;
27            leaderI(a,i);
28        }
29    }
30    private static int leaderJ(int[]a,int i,int j,int count)
31    {
32        if(j<a.length)
33        {
34            if(a[i]<a[j])
35            {
36                count++;
37            }
38        }
39        j++;
40        return leaderJ(a,i,j,count);
41    }
42    else
43    {
44        return count;
45    }
46 }
```

eQuestions

QName	Action
array contain specific value	Load
removeASpecifiedCharacterFrom AString	Load
check equality of two array	Load
find leader element in array	Load
SecondMostFrequentChar	Load
find second max	Load
FindFrequencyOfeachCharacter	Load
GivenStringStartsWithTheContentsOfAnotherString	Load
remove duplicate	Load
stringContainsOnlyDigits	Load
printMax EElement of Matrix array	Load
avg of inner element of 2D array	Load
compareAGivenStringToTheSpecifiedcharacterSequence	Load
basic sort	Load
cipher exchanges	Load
interleaving of two array	Load
sortNumbersInAString.	Load
Find seedal point	Load

ProblemStatement

Write a program to remove a specified character from a given string.
i/p :

```
String s="Aabcd";
Char ch='a';
o/p :Abcd
```

RestrictedKeyWordsInCode: for,while

yourCode

```
1 public class MyClass
2 {
3     //StartOfMainMethod
4     public static void main(String[] args)
5     {
6         String s1="Aabcd";
7         char ch='a';
8         myCode(s1,ch);
9     }
10    //EndOfMainMethod
11    private static void myCode(String s1,char ch)
12    {
13        String s2="";
14        s2=loopi(0,s1,s2,ch);
15        System.out.println(s2);
16    }
17    private static String loopi(int i,String s1,String s2,char ch)
18    {
19        if(i<s1.length())
20        {
21            if(s1.charAt(i)!=ch)
22            {
23                s2=s2+s1.charAt(i);
24            }
25            i++;
26        }
27        s2=loopi(i,s1,s2,ch);
28    }
29    return s2;
30 }
```

HideQuestions

SN	QName	Action
1	array contain specific value	Load
2	removeASpecifiedCharacterFrom AString	Load
3	check equality of two array	Load
4	find leader element in array	Load
5	SecondMostFrequentChar	Load
6	find second max	Load
7	FindFrequencyOfeachCharacter	Load
8	GivenStringStartsWithTheContentsOfAnotherString	Load
9	remove duplicate	Load
10	stringContainsOnlyDigits	Load
11	printMax EElement of Matrix array	Load
12	avg of inner element of 2D array	Load
13	compareAGivenStringToTheSpecifiedcharacterSequence	Load
14	basic sort	Load
15	cipher exchanges	Load
16	interleaving of two array	Load
17	sortNumbersInAString.	Load
18	Find sadal point	Load
19	sumOf TheDigitsPresentInTheGivenString	Load
20	removeDuplicateEleFromString	Load

yourCode

```
1 public class MyClass
2 {
3     //StartOfMainMethod
4     public static void main(String[] args)
5     {
6         int[] a=
7         {
8             33,7,90,20,5,50,40
9         }
10    ;
11    int ele=7;
12    myCode(a,ele);
13 }
14 //EndOfMainMethod
15 private static void myCode(int[] a,int ele)
16 {
17     int cnt=0;
18     cnt=loopi(a,0,ele,cnt);
19     if(cnt==0)
20     {
21         System.out.println("not found");
22     }
23 }
24 private static int loopi(int[]a,int i, int ele,int cnt)
25 {
26     if(i<a.length)
27     {
28         if(a[i]==ele)
29         {
30             System.out.println("found");
31             cnt++;
32             return cnt;
33         }
34         i++;
35     }
36     return loopi(a,i,ele,cnt);
37 }
38 else
39 return cnt;
40 }
```

ideQuestions

QName	Action
array contain specific value	Load
removeASpecifiedCharacterFrom AString	Load
check equality of two array	Load
find leader element in array	Load
SecondMostFrequentChar	Load
find second max	Load
FindFrequencyOfeachCharacter	Load
GivenStringStartsWithTheContentsOfAnotherString	Load
remove duplicate	Load
stringContainsOnlyDigits	Load
printMax EElement of Matrix array	Load
avg of inner element of 2D array	Load
compareAGivenStringToTheSpecifiedcharacterSequence	Load
basic sort	Load
cipher exchanges	Load
interleaving of two array	Load
sortNumbersInAString.	Load
Find sadal point	Load
sumOf TheDigitsPresentInTheGivenString	Load

yourCode

```

1 public class MyClass
2 {
3     //StartOfMainMethod
4     public static void main(String[] args)
5     {
6         int[] a= {1,2,3,4,5};
7         int[] b= {1,2,3,5,4};
8         myCode(a,b);
9     }
10    //EndOfMainMethod
11    private static void myCode(int[] a,int[] b)
12    {
13        if(a.length==b.length)
14        {
15            int count=0;
16            int count1=loopI(a,0,b,count);
17            if(count1==0)
18                System.out.println("same");
19            else
20            {
21                System.out.println("not same");
22            }
23        }
24        private static int loopI(int[]a,int i,int []b,int count)
25        {
26            if(i<a.length)
27            {
28                if(a[i]!=b[i])
29                {
30                    count++;
31                }
32                i++;
33            }
34            return loopI(a,i,b,count);
35        }
36    }
37}
38
39

```

pattern9	Load
Binary Search	Load
SecondMostFrequentChar	Load
pattern24	Load
printAllTheDuplicatesInTheInputString	Load
FindFrequencyOfeachCharacter	Load
implementTrim Method	Load
GivenStringStartsWithTheContentsOfAnotherString	Load
pattern8	Load
find second max	Load
implement splitmethod	Load
find unique pair	Load
pattern15	Load
Strictly Inc And Dec	Load
GivenStringEndsWithTheContentsOfAnotherString	Load
printMax Element of Matrix array	Load
find prime no between given range	Load
Addition of two matrix	Load
sumOf TheDigitsPresentInTheGivenString	Load
Write a program toCompareTwoStringsLexicographically.	Load
implement indexOfMethod	Load
remove specific element	Load
check equality of two array	Load
remove1stFrom2nd	Load

```

1 public class MyClass
{
2     //StartOfMainMethod
3     public static void main(String[] args)
4     {
5         int[] a=
6         {
7             33,7,90,20,5,50,40
8         }
9         ;
10        int ele=78;
11        myCode(a,ele);
12    }
13    //EndOfMainMethod
14    private static void myCode(int[] a,int ele)
15    {
16        //WriteCode Here
17        loopi(a,ele,0);
18    }
19    private static void loopi(int []a,int ele,int i)
20    {
21        if(i<a.length)
22        {
23            if(a[i]!=ele)
24            {
25                System.out.print(a[i]+" ");
26            }
27            loopi(a,ele,++i);
28        }
29    }
30 }
31

```

AlignYourCode (ctrl+a) | UpdateOnServer | Run | Submit

OutPut

Hide Questions

SN	QName	Action
1	avg non diagonal	Load
2	pattern9	Load
3	Binary Search	Load
4	SecondMostFrequentChar	Load
5	pattern24	Load
6	printAllTheDuplicatesInTheInputString	Load
7	FindFrequencyOfeachCharacter	Load
8	implementTrim Method	Load
9	GivenStringStartsWithTheContentsOfAnotherString	Load
10	pattern8	Load
11	find second max	Load
12	implement splitmethod	Load
13	find unique pair	Load
14	pattern15	Load
15	Strictly Inc And Dec	Load
16	GivenStringEndsWithTheContentsOfAnotherString	Load
17	printMax Element of Matrix array	Load
18	find prime no between given range	Load
19	Addition of two matrix	Load
20	sumOf TheDigitsPresentInTheGivenString	Load
21	Write a program to CompareTwoStringsLexicographically.	Load
22	implement indexOfMethod	Load
23	remove specific element	Load
24	check equality of two array	Load
25	remove 1stFrom2nd	Load

yourCode

```
1 public class MyClass
2 {
3     //StartOfMainMethod
4     public static void main(String[] args)
5     {
6         String s1 ="abcdabcd";
7         myCode(s1);
8     }
9     //EndOfMainMethod
10    private static void myCode(String s1)
11    {
12        //write code here
13        int count=0;
14        String s2="";
15
16        for(int i=0; i<s1.length();i++)
17        {
18            for(int j=i+1;j<s1.length();j++)
19            {
20                if(s1.charAt(i)==s1.charAt(j))
21                {
22                    s2=s2+s1.charAt(i);
23                    count++;
24                    break;
25                }
26            }
27        }
28        System.out.println(count);
29        System.out.println(s2);
30    }
31
32 }
```

AlignYourCode (ctrl+a) UpdateOnServer Run Submit
OutPut

HideQuestions

SN	QName	Action
1	avg non digonal	Load
2	pattern9	Load
3	Binary Search	Load
4	SecondMostFrequentChar	Load
5	pattern24	Load
6	printAllTheDuplicatesInTheInputString	Load
7	FindFrequencyOfeachCharacter	Load
8	implementTrim Method	Load
9	GivenStringStartsWithTheContentsOfAnotherString	Load
10	pattern8	Load
11	find second max	Load
12	implement splitmethod	Load
13	find unique pair	Load
14	pattern15	Load
15	Strictly Inc And Dec	Load
16	GivenStringEndsWithTheContentsOfAnotherString	Load
17	printMax EElement of Matrix array	Load
18	find prime no between given range	Load
19	Addition of two matrix	Load
20	sumOf TheDigitsPresentInTheGivenString	Load
21	Write a program toCompareTwoStringsLexicographically.	Load
22	implement indexOfMethod	Load
23	remove specific element	Load
24	check equality of two array	Load
25	remove1stFrom2nd	Load

yourCode

```
1 public class MyClass
2 {
3     //StartOfMainMethod
4     public static void main(String[] args)
5     {
6         String s1 ="abcd";
7         String s2 ="abCd";
8         myCode(s1,s2);
9     }
10    //EndOfMainMethod
11    private static void myCode(String s1,String s2)
12    {
13        //write code here
14        int i=0,cnt=0;
15        cnt=loopi(0,cnt,s1,s2);
16        if(cnt==s1.length() || cnt==s2.length())
17        {
18            System.out.println(s1.length()-s2.length());
19        }
20    }
21    private static int loopi(int i,int cnt,String s1,String s2)
22    {
23        if(i<s1.length() && i<s2.length())
24        {
25            if(s1.charAt(i) == s2.charAt(i))
26            {
27                cnt++;
28            }
29            else
30            {
31                System.out.println(s1.charAt(i) - s2.charAt(i));
32                return cnt;
33            }
34        }
35        i++;
36        cnt=loopi(i,cnt,s1,s2);
37    }
38 }
39 }
```

AllignYourCode (ctrl+a) | UpdateOnServer | Run | Submit

3	implement Trim Method	Load
9	Given String Starts With The Contents Of Another String	Load
10	pattern8	Load
11	find second max	Load
12	implement splitmethod	Load
13	find unique pair	Load
14	pattern15	Load
15	Strictly Inc And Dec	Load
16	Given String Ends With The Contents Of Another String	Load
17	printMax Element of Matrix array	Load
18	find prime no between given range	Load
19	Addition of two matrix	Load
20	sumOf The Digits Present In The Given String	Load
21	Write a program to Compare Two Strings Lexicographically.	Load
22	implement indexOfMethod	Load
23	remove specific element	Load
24	check equality of two array	Load
25	remove 1st From 2nd	Load

```

private static void myCode(int[] a)
{
    //WriteCode Here
    inc=new int[a.length];
    dec=new int[a.length];
    if(a[0]>a[1])
    {
        dec[0]=a[0];
        inc[0]=a[1];
        decele=a[0];
        incele=a[1];
    }
    else if(a[0]<a[1])
    {
        dec[0]=a[1];
        inc[0]=a[0];
        decele=a[1];
        incele=a[0];
    }
    else
    {
        dec[0]=a[1];
        inc[0]=a[1];
        decele=a[1];
        incele=a[1];
    }
    loopi(2,a);
    loopinc(0,incindex);
    System.out.println();
    loopdec(0,decindex);
}

private static void loopdec(int i, int decindex)
{
    if(i<decindex)
    {
        System.out.print(dec[i]+" ");
        i++;
        loopdec(i,decindex);
    }
}
static void loopinc(int i,int incindex)
{
    if(i<incindex)
    {
        System.out.print(inc[i]+" ");
        i++;
        loopinc(i,incindex);
    }
}
static void loopi(int i,int [] a)
{
    if(i<a.length)
    {
        if(a[i]>incele)
        {
            inc[incindex++]=a[i];
            incele=a[i];
        }
        else if(a[i]<decele)
        {
            dec[decindex++]=a[i];
            decele=a[i];
        }
        else
        {
            System.out.println(-1);
            System.exit(0);
        }
        i++;
        loopi(i,a);
    }
}
}

```

SecondMostFrequentChar

Pattern 4

Pattern 5

PrintAllTheDuplicatesInTheInputString

PrintEachOccurrenceCharacter

ImplementTrimMethod

GivenStringStartsWithTheContentsOfAnotherString

10 Factorial

11 Find second max

12 Implement splitmethod

13 Find unique pair

14 pattern15

15 Swap, Inc And Dec

16 GivenStringEndsWithTheContentsOfAnotherString

17 print4th Element of Matrix array

18 Find prime no between given range

19 Addition of two matrix

20 sumOfTheDigitsPresentInTheGivenString

21 Write a program to compare TwoStringsLexicographically.

22 Implement indexOfMethod

23 Print a specific element

24 Comparison of two array

25 Reverse a string2nd

AlignYourCode (ctrl+a)

UpdateOnServer

Run

Submit

5, 1, 3, 6, 8, 2, 9, 3, 10

;

myCode(a);

});

//EndofMainMethod

static int incindex=1,decindex=1,incle=0,decele=0,inc[],dec[];

private static void myCode(int[] a)

//WriteCode Here

inc=new int[a.length];

dec=new int[a.length];

if(a[0]>a[1])

{

dec[0]=a[0];

inc[0]=a[1];

decele=a[0];

incele=a[1];

}

else if(a[0]<a[1])

{

dec[0]=a[1];

inc[0]=a[0];

decele=a[1];

incele=a[0];

incle=a[1];

decele=a[0];

incele=a[1];

decele=a[1];

incele=a[0];

Type here to search



32C

ideQuestions	
QName	Action
find unique triplets	Load
StringArraySortByLength	Load
lowestFrequencyOfStringCharacter	Load
StringArraySortByLength	Load
pattern16	Load
printMax Element of Matrix array	Load
Selection sort	Load
implementTrim Method	Load
sortNumbersInAString.	Load
pattern19	Load
1 Write a program toCompareTwoStringsLexicographically.	Load
12 Strictly Inc And Dec	Load
13 replaceDwithF	Load
14 find unique pair	Load
15 GetTheCharacterAtTheGivenIndex	Load
16 find leader element in array	Load
17 Write a program toCompareTwoStringsLexicographically.	Load
18 remove1stFrom2nd	Load
19 implement indexOfMethod	Load
20 lowestFrequencyOfStringCharacter	Load
21 FInd unique pair given Num	Load
22 stringContainsOnlyDigits	Load
23 pattern8	Load
24 find second max	Load
25 reverseEveryWordInAString	Load

```

5 //StartOfMainMethod
6 public static void main(String[] args)
7 {
8     String s1 = "welcome to hef";
9     myCode(s1);
10 }
11 //EndOfMainMethod
12 private static void myCode(String s1)
13 {
14     //WriteCode Here
15     String sa[]={s1.split(" ")};  

16     loopi(sa,0);
17     print(sa,0);
18 }
19 private static void print(String sa[],int i)
20 {
21     if(i<sa.length)
22     {
23         System.out.print(sa[i]+" ");
24         i++;
25         print(sa,i);
26     }
27 }
28 private static void loopi(String sa[],int i)
29 {
30     if(i<sa.length)
31     {
32         loopj(sa,i,i+1);
33         i++;
34         loopi(sa,i);
35     }
36 }
37 private static void loopj(String sa[],int i,int j)
38 {
39     if(j<sa.length)
40     {
41         if(sa[i].length()>sa[j].length())
42         {
43             String temp=sa[i];
44             sa[i]=sa[j];
45             sa[j]=temp;
46         }
47         j++;
48         loopj(sa,i,j);
49     }
50 }

```

HideQuestions

SN	QName	Action
1	find unique triplets	Load
2	StringArraySortByLength	Load
3	lowestFrequencyOfStringCharacter	Load
4	StringArraySortByLength	Load
5	pattern16	Load
6	printMax EElement of Matrix array	Load
7	Selection sort	Load
8	implementTrim Method	Load
9	sortNumbersInAString.	Load
10	pattern19	Load
11	Write a program toCompareTwoStringsLexicographically.	Load
12	Strictly Inc And Dec	Load
13	replaceDwithF	Load
14	find unique pair	Load
15	GetTheCharacterAtTheGivenIndex	Load
16	find leader element in array	Load
17	Write a program toCompareTwoStringsLexicographically.	Load
18	remove1stFrom2nd	Load
19	implement indexOfMethod	Load
20	lowestFrequencyOfStringCharacter	Load

Write a program to sort in ascending and descending order by length of the given array of strings.

i/p :
 String s1 = "welcome to hef";
 o/p :
 to hef welcome

yourCode

```

1 import java.util.Arrays;
2 import java.util.Iterator;
3 public class MyClass
4 {
5     //StartOfMainMethod
6     public static void main(String[] args)
7     {
8         String s1 = "welcome to hef";
9         myCode(s1);
10    }
11    //EndOfMainMethod
12    private static void myCode(String s1)
13    {
14        //WriteCode Here
15        String sa[]={s1.split(" "});
16        for(int i=0;i<sa.length;i++)
17        {
18            for(int j=i+1;j<sa.length;j++)
19            {
20                if(sa[i].length()>sa[j].length())
21                {
22                    String temp=sa[i];
23                    sa[i]=sa[j];
24                    sa[j]=temp;
25                }
26            }
27        }
28        for(int i=0;i<sa.length;i++)
29        {
30            System.out.print(sa[i]+" ");
31        }
32    }
33 }
34 }
```

HideQuestions

SN	QName	Action	
1	import java.util.Arrays;		
2	import java.util.Iterator;		
3	public class MyClass {		
4	//StartOfMainMethod		
5	public static void main(String[] args)		
6	{		
7	String s1 = "welcome to hefshine";		
8	myCode(s1);		
9	}		
10	//EndOfMainMethod		
11	private static void myCode(String s1)		
12	{		
13	//WriteCode Here		
14	String [] sa=s1.split(" ");		
15	for(int i=sa.length-1;i>=0;i--)		
16	{		
17	System.out.print(sa[i]+" ");		
18	}		
19			
20			
21			
22			
23			
24			
25			
26			
27			

HideQuestions		
N	QName	Action
	MaximumFrequencyOfStringCharacter	Load
	Pattern40	Load
	FindFrequencyOfeachCharacter	Load
	stringContainsOnlyDigits	Load
	StringConvertToUpperCase	Load
	find second max	Load
	find leader element in array	Load
	Sum And Avg of Array element.	Load
	transpose of matrix	Load
0	lowestFrequencyOfStringCharacter	Load
1	reverse Array using third variable	Load
2	least of subArray	Load
3	sum of digonal	Load
4	sortArrayLikeDictionary	Load
5	check equality of two array	Load
6	lowestFrequencyOfStringCharacter	Load
7	Pattern38	Load
8	Pattern24	Load
9	finLargest & SmallestWordIn AString.	Load
20	find unique pair	Load

```

1 import java.util.Arrays;
2 import java.util.Iterator;
3 public class MyClass
4 {static int max=Integer.MAX_VALUE;
5 static int index=0;
6     //StartOfMainMethod
7     public static void main(String[] args)
8     {
9         String s1 = "welcome";
10        myCode(s1);
11    }
12     //EndOfMainMethod
13     private static void myCode(String s1)
14     {
15         loopi(s1,0);
16         System.out.println(max);
17         System.out.println(s1.charAt(index));
18     }
19     private static void loopi(String s1,int i)
20     {
21         if(i<s1.length())
22         {
23             int count=0;
24             count=loopj(s1,i,0,count);
25             if(count<max)
26             {
27                 max=count;
28                 index=i;
29             }
30             i++;
31             loopi(s1,i);
32         }
33     }
34     private static int loopj(String s1,int i,int j,int count)
35     {
36         if(j<s1.length())
37         {
38             if(s1.charAt(i)==s1.charAt(j))
39             {
40                 count++;
41             }
42             j++;
43             return loopj(s1,i,j,count);
44         }
45         else
46         return count;
47     }
48 }
```

4	count the number of words in String	Load	34	}
5	longest palindromic substring	Load	35	;
6	sortArrayLikeDictionary	Load	36	int[][] c=new int[a.length][a[0].length];
7	removeDuplicatesPrint unique count	Load	37	}
8	convert String to an integer	Load	38	//EndOfMainMethod
9	Strictly Inc And Dec	Load	39	private static void add(int[][] a, int[][] b, int[][] c)
10	least of subArray	Load	40	{
11	find unique pair	Load	41	//code here
12	find second max	Load	42	addloopi(a,b,c,0);
13	check if two strings are rotations	Load	43	printi(c,0);
14	FindFrequencyOfEachCharacter	Load	44	}
15	lowestFrequencyOfStringCharacter	Load	45	private static void printi(int [][]c,int i)
16	Addition of two matrix	Load	46	{
17	Write a program to Compare Two Strings Lexicographically.	Load	47	if(i<c.length)
18	array contain specific value	Load	48	printj(c,i,0);
19	remove specific element	Load	49	System.out.println();
20	compareAGivenStringToTheSpecifiedcharacterSequence	Load	50	printi(c,++i);
			51	}
			52	}
			53	}
			54	}
			55	private static void printj(int[][]c,int i,int j)
			56	{
			57	if(j<c.length)
			58	{
			59	System.out.print(c[i][j]+" ");
			60	printj(c,i,++j);
			61	}
			62	}
			63	}
			64	private static void addloopi(int [][]a,int [][]b,int [][]c,int i)
			65	{
			66	if(i<c.length)
			67	{
			68	addloopj(a,b,c,i,0);
			69	addloopi(a,b,c,++i);
			70	}
		j)	71	}
			72	if(j<a.length)
			73	{
			74	c[i][j]=a[i][j]+b[i][j];
			75	addloopj(a,b,c,i,++j);
			76	}
			77	}
			78	
			79	}
			80	}

SN	QName	Action	Line No.
1	SecondMostFrequentChar	Load	3
2	removeLastFrom2nd	Load	4
3	MaximumFrequencyOfStringCharacter	Load	5
4	count the number of words in String	Load	6
5	longest palindromic substring	Load	7
6	sortArrayLikeDictionary	Load	8
7	removeduplicateprint unique count	Load	9
8	convert String to an integer	Load	10
9	Strictly Inc And Dec	Load	11
10	least of subArray	Load	12
11	find unique pair	Load	13
12	find second max	Load	14
13	check if two strings are rotations	Load	15
14	FindFrequencyOfEachCharacter	Load	16
15	lowestFrequencyOfStringCharacter	Load	17
16	Addition of two matrix	Load	18
17	Write a program toCompareTwoStringsLexicographically.	Load	19
18	array contain specific value	Load	20
19	remove specific element	Load	21
20	compareAGivenStringToTheSpecifiedcharacterSequence	Load	22

```

public class MyClass
{
    //StartOfMainMethod
    public static void main(String[] args)
    {
        String s1="hs i am a boy d";
        myCode(s1);
    }
    //EndOfMainMethod
    private static void myCode(String s1)
    {
        //write code here
        s1=s1.trim();
        int count=0;
        for(int i=0;i<s1.length();i++)
        {
            if(s1.charAt(i)==' ')
            {
                count++;
            }
        }
        if(count==s1.length())
        {
            System.out.println(0);
        }
        else
        {
            System.out.println(count+1);
        }
    }
}

```

HideQuestions

1	public class MyClass	{
2	//StartOfMainMethod	
3	public static void main(String[] args)	{
4	String s1="hs i am a boy d";	
5	myCode(s1);	}
6	String s1="hs i am a boy d";	
7	myCode(s1);	}
8	//EndOfMainMethod	
9	private static void myCode(String s1)	{
10	//write code here	
11	int st=0;	
12	int e=0;	
13	for(int i=0;i<s1.length();i++)	
14	{	
15	if(s1.charAt(i)!=' ')	
16	{	
17	st=i;	
18	}	
19	break;	
20	}	
21	}	
22	for(int i=s1.length()-1;i>0;i--)	
23	{	
24	if(s1.charAt(i)!=' ')	
25	{	
26	e=i;	
27	break;	
28	}	
29	}	
30	}	
31	String x="";	
32	for(int i=st;i<=e;i++)	
33	x+=s1.charAt(i);	
34	String[] s=x.split(" ");	
35	System.out.println(s.length);	
36	}	
37	}	
38		
1	removeLastFrom2nd	Action
2	SecondMostFrequentChar	Load
3	lowestFrequencyOfStringCharacter	Load
4	sortArrayLikeDictionary	Load
5	Write a program to Compare Two Strings Lexicographically.	Load
6	remove specific element	Load
7	FindFrequencyOfEachCharacter	Load
8	Addition of two matrix	Load
9	least of subArray	Load
10	removeuplicateprint unique count	Load
11	MaximumFrequencyOfStringCharacter	Load
12	count the number of words in String	Load
13	convert String to an integer	Load
14	array contain specific value	Load
15	longest palindromic substring	Load
16	check if two strings are rotations	Load
17	find unique pair	Load
18	Strictly Inc And Dec	Load
19	find second max	Load
20	compareAGivenStringToTheSpecifiedcharacterSequence	Load

HideQuestions

SN	QName	Action
1	SecondMostFrequentChar	Load
2	remove1stFrom2nd	Load
3	MaximumFrequencyOfStringCharacter	Load
4	count the number of words in String	Load
5	longest palindromic substring	Load
6	sortArrayLikeDictionary	Load
7	removeduplicateprint unique count	Load
8	convert String to an integer	Load
9	Strictly Inc And Dec	Load
10	least of subArray	Load
11	find unique pair	Load
12	find second max	Load
13	check if two strings are rotations	Load
14	FindFrequencyOfeachCharacter	Load
15	lowestFrequencyOfStringCharacter	Load
16	Addition of two matrix	Load
17	Write a program toCompareTwoStringsLexicographically.	Load
18	array contain specific value	Load
19	remove specific element	Load
20	compareAGivenStringToTheSpecifiedcharacterSequence	Load

```

2 import java.util.Iterator;
3 public class MyClass
{
4     static int max=0;
5     static int secmax=0;
6     static int count=0;
7     static int index=0;
8     //StartOfMainMethod
9     public static void main(String[] args)
10    {
11        String s1 = "abcad";
12        myCode(s1);
13    }
14    //EndOfMainMethod
15    private static void myCode(String s1)
16    {
17        //write code here
18        loopi(s1,0);
19        if(secmax==0)
20        {
21            System.out.println(" -1");
22        }
23        else
24        {
25            System.out.println(s1.charAt(index)+" "+secmax);
26        }
27    }
28    private static void loopi(String s1,int i)
29    {
30        if(i<s1.length())
31        {
32            count=0;
33            loopj(s1,i,0);
34            if(count>max)
35            {
36                max=count;
37            }
38            if(count!=max&&count>secmax)
39            {
40                secmax=count;
41                index=i;
42            }
43            i++;
44            loopi(s1,i);
45        }
46    }
47    private static void loopj(String s1,int i,int j)
48    {
49        if(j<s1.length())
50        {
51            if(s1.charAt(i)==s1.charAt(j))
52            {
53                count++;
54            }
55            j++;
56            loopj(s1,i,j);
57        }
58    }
59

```

```

1 import java.util.Arrays;
2 import java.util.Iterator;
3 public class MyClass
4 {
5     //StartOfMainMethod
6     public static void main(String[] args)
7     {
8         String s1 = "aabbcc";
9         myCode(s1);
10    }
11    //EndOfMainMethod
12    private static void myCode(String s1)
13    {
14        //Write code here
15        String s="";
16        s=loopi(s1,s,0);
17        loopiunique(s,s1,0);
18    }
19    private static void loopiunique(String s, String s1, int i)
20    {
21        if(i<s.length())
22        {
23            int count=0;
24            count=loopjunique(s1,s,i,0,count);
25            System.out.println(s.charAt(i)+" "+count);
26        }
27    }
28    private static int loopjunique(String s1, String s, int i, int j, int count)
29    {
30        if(j<s1.length())
31        {
32            if(s.charAt(i)==s1.charAt(j))
33            {
34                Load
35                Load
36                Load
37                Load
38                Load
39                Load
40                Load
41                Load
42                Load
43                Load
44                Load
45                Load
46                Load
47                Load
48            }
49        }
50    }
51    private static String loopi(String s1, String s, int i)
52    {
53        if(i<s1.length())
54        {
55            int count=0;
56            count=loopj(s1,i,i-1,count);
57        }
58    }
59}

```

HideQuestions

SN	QName	Action
1	SecondMostFrequentChar	Load
2	removelsfFrom2nd	Load
3	MaximumFrequencyOfStringCharacter	Load
4	count the number of words in String	Load
5	longest palindromic substring	Load
6	sortArrayLikeDictionary	Load
7	removeduplicateprint unique count	Load
8	convert String to an integer	Load
9	Strictly Inc And Dec	Load
10	least of subArray	Load
11	find unique pair	Load
12	find second max	Load
13	check if two strings are rotations	Load
14	FindFrequencyOfeachCharacter	Load

```

5 longest palindromic substring
6 sortArrayLikeDictionary
7 removeduplicateprint unique count
8 convert String to an integer
9 Strictly Inc And Dec
10 least of subArray
11 find unique pair
12 find second max
13 check if two strings are rotations
14 FindFrequencyOfeachCharacter
15 lowestFrequencyOfStringCharacter
16 Addition of two matrix
17 Write a program toCompareTwoStringsLexicographically.
18 array contain specific value
19 remove specific element
20 compareAGivenStringToTheSpecifiedcharacterSequence

Load 32
Load 33
Load 34
Load 35
Load 36
Load 37
Load 38
Load 39
Load 40
Load 41
Load 42
Load 43
Load 44
Load 45
Load 46
Load 47
Load 48
Load 49
Load 50
Load 51
Load 52
Load 53
Load 54
Load 55
Load 56
Load 57
Load 58
Load 59
Load 60
Load 61
Load 62
Load 63
Load 64
Load 65
Load 66
Load 67
Load 68
Load 69
Load 70
Load 71
Load 72
Load 73
Load 74

if(s.charAt(i)==s1.charAt(j))
{
    count++;
}
j++;
return loopjunique(s1,s,i,j,count);
}

private static String loopj(String s1,String s,int i)
{
    if(i<s1.length())
    {
        int count=0;
        count=loopj(s1,i,i-1,count);
        if(count==0)
        {
            s=s+s1.charAt(i);
        }
        i++;
        return loopj(s1,s,i);
    }
    else
    {
        return s;
    }
}

private static int loopj(String s1,int i,int j,int count)
{
    if(j>=0)
    {
        if(s1.charAt(i)==s1.charAt(j))
        {
            count++;
        }
        j--;
        return loopj(s1,i,j,count);
    }
    else
    {
        return count;
    }
}

```

[HideQuestions](#)

SN	QName	Action
1	SecondMostFrequentChar	Load
2	remove1stFrom2nd	Load
3	MaximumFrequencyOfStringCharacter	Load
4	count the number of words in String	Load
5	longest palindromic substring	Load
6	sortArrayLikeDictionary	Load
7	removeduplicateprint unique count	Load
8	convert String to an integer	Load
9	Strictly Inc And Dec	Load
10	least of subArray	Load
11	find unique pair	Load
12	find second max	Load
13	check if two strings are rotations	Load
14	FindFrequencyOfeachCharacter	Load
15	lowestFrequencyOfStringCharacter	Load
16	Addition of two matrix	Load
17	Write a program toCompareTwoStringsLexicographically.	Load
18	array contain specific value	Load
19	remove specific element	Load
20	compareAGivenStringToTheSpecifiedcharacterSequence	Load

```
1 public class MyClass
2 {
3     //StartOfMainMethod
4     public static void main(String[] args)
5     {
6         int[] a=
7         {
8             33,7,90,20,5,50,40
9         }
10    ;
11    int ele=7;
12    myCode(a,ele);
13 }
14 //EndOfMainMethod
15 private static void myCode(int[] a,int ele)
16 {
17     //WriteCode Here
18     int count=0;
19     int i=0;
20     count=loopi(i,a,ele,count);
21     if(count==0)
22     {
23         System.out.print("not found");
24     }
25 }
26 private static int loopi(int i,int[] a,int ele,int count)
27 {
28     if(i<a.length)
29     {
30         if(a[i]==ele)
31         {
32             count++;
33             System.out.print("found");
34         }
35         i++;
36     }
37     return loopi(i,a,ele,count);
38 }
39 else
40 {
41 }
42 }
```

HideQuestions

SN	QName	Action
1	FindFrequencyOfEachCharacter	Load
2	compareAGivenStringToTheSpecifiedcharacterSequence	Load
3	longest palindromic substring	Load
4	SecondMostFrequentChar	Load
5	removeduplicateprint unique count	Load
6	least of subArray	Load
7	Addition of two matrix	Load
8	MaximumFrequencyOfStringCharacter	Load
9	Write a program toCompareTwoStringsLexicographically.	Load
10	convert String to an integer	Load
11	find second max	Load
12	sortArrayLikeDictionary	Load
13	Strictly Inc And Dec	Load
14	check if two strings are rotations	Load
15	remove specific element	Load
16	remove1stFrom2nd	Load
17	lowestFrequencyOfStringCharacter	Load
18	count the number of words in String	Load
19	array contain specific value	Load
20	find unique pair	Load