

```
1 import java.util.Scanner;
2 class Calculator
3 {
4     int add(int... val)
5     {
6         int sum=0;
7         for(int a: val)
8         {
9             sum=sum+a;
10        }
11        return sum;
12    }
13    int sub(int... val)
14    {
15        int diff=0;
16        for(int a: val)
17        {
18            diff=diff-a;
19            if(a==val[0])
20            {
21                diff=-diff;
22            }
23        }
24        return diff;
25    }
26    int mul(int... val)
27    {
28        int mul=1;
29        for(int a: val)
30        {
31            mul=mul*a;
32        }
33        return mul;
34    }
35    double div(int... val)
36    {
37        double div=1;
38        for(int a: val)
39        {
40            div=div/a;
41        }
42        return div;
43    }
44 }
45 public class Main
46 {
```

```
48 {
49     Calculator c=new Calculator();
50     Scanner sc=new Scanner(System.in);
51     System.out.println("Enter the size of arrays :");
52     int size=sc.nextInt();
53     int arr1[]={};
54     int arr2[]={};
55     char operator[]={'+','-', '*', '/'};
56     System.out.println("Enter first array elements :");
57     for(int i=0;i<size;i++)
58     {
59         arr1[i]=sc.nextInt();
60     }
61     System.out.println("Enter second array elements :");
62     for(int i=0;i<size;i++)
63     {
64         arr2[i]=sc.nextInt();
65     }
66     System.out.println("Choose your operator to perform :");
67     for(int i=0;i<operator.length;i++)
68     {
69         System.out.println(operator[i]);
70     }
71     char op=sc.next().charAt(0);
72     int flag=0;
73     for(int i=0;i<operator.length;i++)
74     {
75         if(op==operator[i])
76         {
77             switch(i)
78             {
79                 case 0 :
80                     System.out.println("Adding two arrays :");
81                     for(int j=0;j<arr1.length;j++)
82                     {
83                         flag=1;
84                         System.out.println(arr1[j]+arr2[j]);
85                     }
86                     break;
87                 case 1 :
88                     System.out.println("Subtracting two arrays :");
89                     for(int j=0;j<arr1.length;j++)
90                     {
91                         flag=1;
92                         System.out.println(arr1[j]-arr2[j]);
93                     }
94                     break;
95             }
96         }
97     }
98 }
```

```
80         System.out.println("Adding");
81         for(int j=0;j<arr1.length;
82         {
83             flag=1;
84             System.out.println(arr
85         }
86         break;
87     case 1 :
88         System.out.println("Subtra
89         for(int j=0;j<arr1.length;
90         {
91             flag=1;
92             System.out.println(arr
93         }
94         break;
95     case 2 :
96         System.out.println("multip
97         for(int j=0;j<arr1.length;
98         {
99             flag=1;
100            System.out.println(arr
101        }
102        break;
103    case 3 :
104        System.out.println("Dividi
105        for(int j=0;j<arr1.length;
106        {
107            flag=1;
108            System.out.println(arr
109        }
110        break;
111    }
112}
113}
114if(flag==0)
115{
116    System.out.println("Invalid operator")
117}
118}
119}
```

X Terminal



Enter the size of arrays :

2

Enter first array elements:

37

36

Enter second array elements:

73

47

Choose your operator to calculate from the option

+

-

*

/

+

Adding elements in both the arrays :

37+73=110

36+47=83

Process finished.

```
1 import java.util.*;
2 class Main
3 {
4     public static boolean isPalindrome(String i, int
5     {
6         if (low >= high) {
7             return true;
8         }
9
10        if (i.charAt(low) != i.charAt(high)) {
11            return false;
12        }
13
14        return isPalindrome(i, low + 1, high - 1);
15    }
16
17    public static void main(String[] args)
18    {
19        String i = "sravan";
20
21        if (isPalindrome(i, 0, i.length() - 1)) {
22            System.out.print("Yes");
23        } else {
24            System.out.print("No");
25        }
26    }
27 }
28 }
```

X Terminal



No
Process finished.

```
1 import java.io.*;
2
3
4
5 class Sravan
6 {
7
8
9 // Function to count digits
10
11 static int countEvenOdd(int n)
12 {
13
14     int even_count = 0;
15
16     int odd_count = 0;
17
18     while (n > 0)
19     {
20
21         int rem = n % 10;
22
23         if (rem % 2 == 0)
24
25             even_count++;
26
27         else
28
29             odd_count++;
30
31         n = n / 10;
32
33     }
34
35
36     System.out.println ("Even count : " +
37
38                         even_count);
39
40     System.out.println ("Odd count : " +
41
42                         odd_count);
43
44     if (even_count % 2 == 0 &&
45
46         odd_count % 2 != 0)
```

```
36     System.out.println ( "Even count : " +
37                                         even_count);
38
39     System.out.println ( "Odd count : " +
40                                         odd_count);
41
42     if (even_count % 2 == 0 &&
43         odd_count % 2 != 0)
44         return 1;
45
46     else
47         return 0;
48 }
49
50
51
52
53 } // Driver Code
54
55
56
57 public static void main (String[] args)
58 {
59     int n;
60
61     n = 134732;
62
63     int t = countEvenOdd(n);
64
65
66
67     if (t == 1)
68         System.out.println ( "YES" );
69
70     else
71         System.out.println( "NO" );
72
73
74
75
76
77
78
79
80
81 }
```

X Terminal



Even count : 2

Odd count : 4

NO

Process finished.

```
3  
4 import java.util.Arrays;  
5  
6  
7  
8 public class GFG  
9 {  
10     // A function to implement bubble sort  
11  
12     static void bubbleSort(int arr[], int n)  
13     {  
14         // Base case  
15         if (n == 1)  
16             return;  
17  
18  
19         // One pass of bubble sort. After  
20         // this pass, the largest element  
21         // is moved (or bubbled) to end.  
22  
23  
24         for (int i=0; i<n-1; i++)  
25             if (arr[i] > arr[i+1])  
26                 {  
27                     // swap arr[i], arr[i+1]  
28  
29                     int temp = arr[i];  
30  
31                     arr[i] = arr[i+1];  
32  
33                     arr[i+1] = temp;  
34  
35                 }  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46 }
```

```
39         int temp = arr[i];
40
41         arr[i] = arr[i+1];
42
43         arr[i+1] = temp;
44
45     }
46
47
48
49     // Largest element is fixed,
50
51     // recur for remaining array
52
53     bubbleSort(arr, n-1);
54
55 }
56
57
58
59 // Driver Method
60
61 public static void main(String[] args)
62 {
63
64     int arr[] = {64, 34, 25, 12, 22, 11, 90};
65
66
67
68     bubbleSort(arr, arr.length);
69
70
71
72
73     System.out.println("Sorted array : ");
74
75     System.out.println(Arrays.toString(arr));
76
77 }
78 }
```

X Terminal



```
Sorted array :  
[11, 12, 22, 25, 34, 64, 90]
```

```
Process finished.
```