

SAVEETHA SCHOOL OF ENGINEERING
SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL SCIENCES
INSTITUTE OF PLACEMENT AND TRAINING
CSA09 –JAVA PROGRAMMING

Name: V. Balaram Krishna
Reg.No: 192324049

1. Write a program to calculate the factorial of number using recursive function.

Sample Input & Output:

Enter the value of n: 6

Sample Input & Output:

The factorial of 6 is: 720

Test cases:

1. N = 0
2. N = -5
3. N = 1
4. N = M
5. N = %

CODE:

```
factorialrecursion.java > ...
1  import java.util.Scanner;
2  public class factorialrecursion {
3      static int fact(int num){
4          if(num==0||num==1)
5              return 1;
6          else
7              return num*fact(num-1);
8      }
9
10     Run | Debug | Run main | Debug main
11     public static void main(String[] args) {
12         int num;
13         Scanner input=new Scanner(System.in);
14         System.out.println("Enter number : ");
15         num=input.nextInt();
16         fact(num);
17         System.out.println("Factorial is : "+fact(num));
18     }
19 }
```

OUTPUT:

```
Enter number :
5
Factorial is : 120
```

2. Write a Program to Find the Nth Largest Number in a array.

Sample Input:

List : {14, 67, 48, 23, 5, 62}

N = 4

Sample Output:

4th Largest number: 23

Test cases:

1. N = 0
2. N = -5
3. N = 1
4. N = M
5. N = %

CODE:

```
J nthlargest.java > Language Support for Java(TM) by Red Hat > nthlargest > main(String[])
1 import java.util.Arrays;
2 import java.util.Scanner;
3
4 public class nthlargest {
5     public static void main(String[] args){
6         int n;
7         Scanner a=new Scanner(System.in);
8         System.out.println("Enter Array Size : ");
9         n=a.nextInt();
10        int arr[]=new int[n];
11        System.out.println("Enter "+n+" Array Elements : ");
12        for(int i=0;i<n;i++){
13            arr[i]=a.nextInt();
14        }
15        System.out.println("Enter for nth largest number : ");
16        int k=a.nextInt();
17        Arrays.sort(arr);
18        System.out.println("Kth Largest num in array is : "+arr[n-k]);
19    }
20
21 }
22 }
```

OUTPUT:

```
Enter Array Size :
6
Enter 6 Array Elements :
14
67
48
23
5
62
Enter for nth largest number :
4
Kth Largest num in array is : 23
```

3. Write a program to convert the Binary to Decimal, Octal

Sample Input:

Given Number: 1101

Sample Output:

Decimal Number: 13

Octal:15

Test cases:

1. 211
2. 11011
3. 22122
4. 111011.011
5. 1010.0101

CODE:

```
J bintodecandoct.java > Language Support for Java(TM) by Red Hat > bintodecandoct > main(String[])
1 import java.util.Scanner;
2 public class bintodecandoct {
3     public static void main(String[] args) {
4         String n;
5         Scanner a=new Scanner(System.in);
6         System.out.print("Enter Binary Value : ");
7         n=a.nextLine();
8         int dec=Integer.parseInt(n,2);
9         String oct=Integer.toOctalString(dec);
10        System.out.println("Decimal Number is : "+dec);
11        System.out.println("Octal Number is : "+oct);
12    }
13 }
14 }
```

OUTPUT:

```
Enter Binary Value : 1101
Decimal Number is : 13
Octal Number is : 15
```

4. Write a program to find the number of special characters in the given statement
Sample Input:
Given statement: Modi Birthday @ September 17, #&\$% is the wishes code for him.
Sample Output:
Number of special Characters: 5

CODE:

```
J countspecial.java > Language Support for Java(TM) by Red Hat > countspecial > main(String[])
1 import java.util.Scanner;
2 public class countspecial {
3     public static void main(String[] args) {
4         String n;
5         Scanner a=new Scanner(System.in);
6         System.out.println("Enter the Sentence : ");
7         n=a.nextLine();
8         char ch;
9         int count=0;
10        for(int i=0;i<n.length();i++){
11            ch=n.charAt(i);
12
13            if(!Character.isLetterOrDigit(ch) && !Character.isWhitespace(ch)){
14                count+=1;
15            }
16        }
17        System.out.println("No of Special Characters are : "+count);
18    }
19 }
20
```

OUTPUT:

```
Enter the Sentence :
modi born on @56 at $india&*
No of Special Characters are : 4
```

5. Write a Program to Remove the Duplicate Items from a array.
Sample Input:

```
Enter the number of elements in array:7
Enter element1:10
Enter element2:20
Enter element3:20
Enter element4:30
Enter element5:40
Enter element6:40
Enter element7:50
```

Sample Output:
Non-duplicate items:
[10, 20, 30, 40, 50]

CODE:

```

J duplicateelements.java > Language Support for Java(TM) by Red Hat > duplicateelements > main(String[])
1 public class duplicateelements {
    Run | Debug | Run main | Debug main
2     public static void main (String[] args) {
3         int arr[] = {10, 20, 20, 30, 30, 40, 50, 50};
4         int n = arr.length;
5         int[] temp = new int[n];
6         int j = 0;
7         for (int i = 0; i < n - 1; i++) {
8             if (arr[i] != arr[i + 1]) {
9                 temp[j++] = arr[i];
10            }
11        }
12        temp[j++] = arr[n - 1];
13        for(int i = 0; i < j; i++) {
14            arr[i] = temp[i];
15        }
16
17        for (int i = 0; i < j; i++) {
18            System.out.print(arr[i] + " ");
19        }
20    }
21 }
22

```

OUTPUT:

```
10 20 30 40 50
```

- Display Multiplication table for 5 and 10 using various stages of life cycle of the thread by generating a suitable code in Java.

Sample Input 5, 10

5 X 1 = 5

5 X 2 = 10

....

10 X 1 = 10

10 X 2 = 20

....

Test Cases:

- 10, 20
- 10, -30
- 0, 0
- SIX, SIX
- 9.8, 9.6

CODE:

```

J twomultiplicationtables.java > ...
1 public class twomultiplicationtables {
    Run | Debug | Run main | Debug main
2     public static void main(String[] args){
3         int n1=5;
4         int n2=10;
5         System.out.println("Multiplication table for "+n1);
6         for(int i=1;i<=10;i++){
7             System.out.println(n1+"*"+i+"="+n1*i);
8         }
9         System.out.println("Multiplication table for "+n2);
10        for(int i=1;i<=10;i++){
11            System.out.println(n2+"*"+i+"="+n2*i);
12        }
13    }
14 }
15

```

OUTPUT:

```
Multiplication table for 5
5*1=5
5*2=10
5*3=15
5*4=20
5*5=25
5*6=30
5*7=35
5*8=40
5*9=45
5*10=50
Multiplication table for 10
10*1=10
10*2=20
10*3=30
10*4=40
10*5=50
10*6=60
10*7=70
10*8=80
10*9=90
10*10=100
```

7. Using the concepts of thread with implementing Runnable interface in Java to generate Fibonacci series.

Sample Input : 5

Sample Output : 0 1 1 2 3

Test Cases

1. 7
2. -10
3. 0
4. EIGHT FIVE
5. 12.65

CODE:

```
fibonacciRunnable.java > Language Support for Java(TM) by Red Hat > fibonacciRunnable > main(String[])
1 public class fibonacciRunnable implements Runnable{
2     int n;
3     public fibonacciRunnable(int n){
4         this.n=n;
5     }
6     public void run(){
7         int a=0,b=1;
8         System.out.println("Fibonacci Series : ");
9         for(int i=0;i<n;i++){
10             System.out.print(a+" ");
11             int sum=a+b;
12             a=b;
13             b=sum;
14         }
15     }
16     public static void main(String[] args) {
17         int n=5;
18         Thread fibonacciThread=new Thread(new fibonacciRunnable(n));
19         fibonacciThread.start();
20     }
21 }
```

OUTPUT:

```
Fibonacci Series :
0 1 1 2 3
```

8. Generate a Java code to find the sum of N numbers using array and throw ArrayIndexOutOfBoundsException when the loop variable beyond the size N.

Sample Input : 5

1 2 3 4 5

Sample Output : 15

Test Cases

1. 4, 10

2. -10
3. 0
4. EIGHT SEVEN
5. 12.68

CODE:

```
exceptionarrayindex.java > Language Support for Java(TM) by Red Hat > exceptionarrayindex > main(String[])
1 public class exceptionarrayindex {
  Run | Debug | Run main | Debug main
2   public static void main(String[] args){
3       int arr[]={1,2,3,4,5};
4       try{
5           for(int i=0;i<=arr.length;i++){
6               system.out.println(arr[i]+" ");
7           }
8       }
9       catch(Exception e){
10          system.out.println("Index Out of Range");
11      }
12  }
13  }
14 }
```

OUTPUT:

```
1
2
3
4
5
Index Out of Range
```

9. Using the concepts of thread with implementing Runnable interface in Java to find whether a given number is prime or not.

Sample Input : 5

Sample Output : 5 is Prime

Sample Output : 15

Test Cases

1. 4
2. -10
3. 0
4. EIGHT SEVEN
5. 11.48

CODE:

```

1 public class primerunnable implements Runnable{
2     private int n;
3     public primerunnable(int n){
4         this.n=n;
5     }
6     public void run(){
7         if(isprime(n)){
8             System.out.println("Prime");
9         }
10        else{
11            System.out.println("Not Prime");
12        }
13    }
14    public Boolean isprime(int n){
15        if(n<2){
16            return false;
17        }
18        else{
19            for(int i=2;i<=Math.sqrt(n);i++){
20                if(n%i==0){
21                    return true;
22                }
23            }
24        }
25        return false;
26    }
27    Run | Debug | Run main | Debug main
28    public static void main(String[] args){
29        int n=29;
30        Thread primethread=new Thread(new primerunnable(n));
31        primethread.start();
32    }
33 }

```

OUTPUT:

Prime

10. Generate a Java code to handle Exceptions such as Arithmetic Exception, ArrayIndexOutOfBoundsException, NullPointerException using Multi-Catch Statements.

NULLPOINTER EXCEPTION:

CODE:

```

1 public class exceptions {
2     Run | Debug | Run main | Debug main
3     public static void main(String[] args) {
4         String str=null;
5         try{
6             if(str.equals("hello")){
7                 System.out.println("same");
8             }
9             else{
10                System.out.println("Not Same");
11            }
12        }
13        catch(NullPointerException e){
14            System.out.println("Nullpointer Exception");
15        }
16    }
17 }

```

OUTPUT:

Nullpointer Exception

ARRAYOUTOFBOUNDS:

CODE:

```

J exceptionarrayindex.java > Language Support for Java(TM) by Red Hat > exceptionarrayindex > main(String[])
1 public class exceptionarrayindex {
    Run | Debug | Run main | Debug main
2     public static void main(String[] args){
3         int arr[]={1,2,3,4,5};
4         try{
5             for(int i=0;i<arr.length;i++){
6                 System.out.println(arr[i]+" ");
7             }
8         }
9         catch(Exception e){
10            System.out.println("Index Out of Range");
11        }
12    }
13 }
14

```

OUTPUT:

```

1
2
3
4
5
Index Out of Range

```

ARITHMETICEXPRESSION:

CODE:

```

J exceptionarithmetic.java > Language Support for Java(TM) by Red Hat > exceptionarithmetic > main(String[])
1 public class exceptionarithmetic {
    Run | Debug | Run main | Debug main
2     public static void main(String[] args) {
3         int a=10;
4         int b=0;
5         try{
6             double avg=a/b;
7             System.out.println("Ans: "+avg);
8         }
9         catch(ArithmeticException e){
10            System.out.println("Can't Divide by zero");
11        }
12    }
13 }
14

```

OUTPUT:

```

Can't Divide by zero

```