Aditya Degree Colleges

Java Online Training Coding Test_8 Key

Date: 30-04-2020

Program – 1:

Counting Bits

Ramu is an enthusiastic student. He knows that in computer terminology byte means 8 bits. Numbers from 0 – 255 can be represented using a byte.

However, all the 8 bits are not required for all the numbers in the range 0- 255. For example 4 in binary is 100 which requires only 3 bits and 9 in binary is 1001 which requires only 4 bits. Now, ramu has been confronted with a problem. He has a list of positive integers which are in between [0,255] and he wants to know exactly how many total bits are required for all those numbers, if they are represented in binary.

Sample Test Cases

Test case 1:

Input

23 3 45 67 100

output

27

Explanation:

the binary representation of the elements given is,

10111 11 101101 1000011 1100100

whose total length i.e bits is 27

Test case 2:

Input:

2 56 12 78 89 45

output:

32

Explanation:

the binary representation of the elements given is,

10 111000 1100 1001110 1011001 101101

whose total length i.e bits is 32

Hidden test cases:

Test case1:

input:

2 56 12 78 8

output:

23

Test case2:
input:
2 56 12 78 8 0
output:
24

Test case3:
input:
255 2 56 12 78 8
output:
31

Test case4:
input:
2 56 12 2 2
output:
16

Source Code:

```
import java.util.*;
import java.io.*;
import java.math.*;
class CountBits{
static int countBits(int item)
{
if(item==0){
return 1;
}
else{
return (int)(Math.log(item)/Math.log(2) + 1);
}
}
public static void main(String[] args) throws IOException
{
Scanner sc=new Scanner(System.in);
int count=0;
String strNums = sc.nextLine();
String[] tokens = strNums.split(" ");
int[] ary = new int[tokens.length];
int i = 0;
for (i = 0; i<tokens.length; ++i){
ary[i] = Integer.parseInt(tokens[i]);
}
for(i=0;i<ary.length;i++){
count+=countBits(ary[i]);
}
System.out.println(count);
}
}
```

Program – 2:

Automorphic Number

Vijaya is known for her talent in maths. Now she has been given a task by her neighbor who is also very good at math. The task is neighbour will give a number she has to square the given number and if the squared number is ending with input number itself then she need to say "YES" otherwise "NO". Help vijaya in completing the task. If the input number is less than 4 she need to say "Invalid Input"

Note: Input number can be any digited number so you need to check for that many digits after squaring the number.

Input:

A single line consists of n digited input number

Constraints:

4<=N<=10000

Output:

A single line either displays "YES" or "NO" or "Invalid Input"

Sample Test Case 1:

Input:

5

Output:

YES

Explaination:

$5^2=25$ ending digit is 5 the input number itself

Sample Test Case 2:

Input:

1

Output:

Invalid Input

Hidden Test Case 1:

Input:

6

Output:

YES

Hidden Test Case 2:

Input:

76

Output:

YES

Hidden Test Case 3:

Input:

7

Output:

NO

Hidden Test Case 4:

Input:

100

Output:

NO

Java Code:

```
import java.util.*;
import java.io.*;
class Automorphic{
public static void main(String[] args) throws IOException
{
Scanner sc=new Scanner(System.in);
int n=sc.nextInt();
if(n<=3){
System.out.print("Invalid Input");
}
else{
int sq_n=n*n;
String str_num = Integer.toString(n);
String square = Integer.toString(sq_n);
if(square.endsWith(str_num))
System.out.println("YES");
else
System.out.println("NO");
}
}
}
```

Or

```
import java.util.*;
import java.io.*;
class Automorphic1{
public static void main(String[] args) throws IOException
{
Scanner sc=new Scanner(System.in);
int n=sc.nextInt();
int flag=0;
if(n<=3){
flag=-1;
}
else{
int sq_n=n*n;
while(n>0){
if(n%10!=sq_n%10){
flag=1;
break;
}
n=n/10;
sq_n=sq_n/10;
}
}
if(flag==0){
System.out.println("YES");
}
else if(flag==-1){
System.out.println("Invalid Input");
}
else{
System.out.println("NO");
}
}
}
```

MCQ_KEY

What is the output of the following program?

```
public class Test
{
    public static void main(String[] args)
    {
        try
        {
            System.out.printf("1");
            int data = 5 / 0;
        }
        catch(ArithmeticException e)
        {
            System.out.printf("2");
            System.exit(0);
        }
        finally
        {
            System.out.printf("3");
        }
        System.out.printf("4");
    }
}
```

a) 12
b) 1234
c) 124
d) 123

ANSWER: A

What is the output of the following program?

```
import java.io.EOFException;
import java.io.IOException;
public class Test
{
    public static void main(String[] args)
    {
        try
        {
            System.out.printf("1");
            int value = 10 / 0;
            throw new IOException();
```

```
        }
        catch(EOFException e)
        {
            System.out.printf("2");
        }
        catch(ArithmeticException e)
        {
            System.out.printf("3");
        }
        catch(NullPointerException e)
        {
            System.out.printf("4");
        }
        catch(IOException e)
        {
            System.out.printf("5");
        }
        catch(Exception e)
        {
            System.out.printf("6");
        }
    }
}
```

a) 1346
b) 136726
c) 136
d) 13

ANSWER:D

Which of these keywords is not a part of exception handling?
a) try
b) finally
c) thrown
d) catch

ANSWER: C

Which of these keywords is used to manually throw an exception?
a) try
b) finally
c) throw
d) catch

ANSWER: C

What will be the output of the following Java program?

```
class exception_handling
{
    public static void main(String args[])
    {
        try
        {
            System.out.print("Hello" + " " + 1 / 0);
        }
        catch(ArithmeticException e)
        {
      System.out.print("World");
        }
    }
}
```

A) Hello
B) World
C) HelloWorld
D) Hello World

ANSWER:B

```
class exception_handling
{
    public static void main(String args[])
    {
        try
        {
            int i, sum;
            sum = 10;
            for (i = -1; i < 3 ;++i)
                sum = (sum / i);
        }
        catch(ArithmeticException e)
        {
      System.out.print("0");
        }
        System.out.print(sum);
    }
}
```

a) 0
b) 05
c) Compilation Error

d) Runtime Error
ANSWER:C

What will be the output of the following Java program?

```
class exception_handling
{
    public static void main(String args[])
    {
        try
       {
            int a, b;
            b = 0;
            a = 5 / b;
            System.out.print("A");
        }
        catch(ArithmeticException e)
        {
      System.out.print("B");
        }
    }
}
```

a) A
b) B
c) Compilation Error
d) Runtime Error
ANSWER:B

What will be the output of the following Java program?

```
class exception_handling
{
    public static void main(String args[])
    {
        try
        {
            int a, b;
            b = 0;
            a = 5 / b;
            System.out.print("A");
        }
        catch(ArithmeticException e)
        {
      System.out.print("B");
```

```
        }
        finally
        {
            System.out.print("C");
        }
      }
   }
```

a) A
b) B
c) AC
d) BC

ANSWER:D

What is the use of final keyword in Java?
A) When a class is made final, a sublcass of it can not be created.
B)When a method is final, it can not be overridden.
C)When a variable is final, it can be assigned value only once.
D)All of the above

ANSWER: D

Output of follwoing Java program

```
class Main {
 public static void main(String args[]){
   final int i;
   i = 20;
   System.out.println(i);
 }
}
```

A)20
B)Compiler Error
C)0
D)1

ANSWER:A

```
class Base {
  public final void show() {
      System.out.println("Base::show() called");
   }
}
class Derived extends Base {
    public void show() {
      System.out.println("Derived::show() called");
```

```
    }
}
class Main {
    public static void main(String[] args) {
        Base b = new Derived();;
        b.show();
    }
}
```

A) Derived::show() called
B) Base::show() called
C) Compiler Error
D) Runtime
ANSWER:C

```
class A implements B
{
    public int methodB(int i)
    {
        return i =+ i * i;
    }
}
 interface B
{
    int methodB(int i);
}

class MainClass
{
    public static void main(String[] args)
    {
        B b = new A();

        System.out.println(b.methodB(2));
    }
}
```

A) 4
B) 2
C) 6
D) 8
ANSWER:A

```
interface A
{
   int i = 111;
}
 class B implements A
{
   void methodB()
   {
      i = 222;
   }
}
```

A) No Error
B) Compile time error with methodB() not in A
C) Compile time error with i value can't be modified
D) Compile time error with i varible is declared as final
ANSWER:C

Which of the following statement is invalid
A) one interface extends more than one interface
B) one class implements more than one interface
C) More than one class implements the same interface
D) one interface implements a class.
ANSWER:D

Which of the following is not a marker interface?
A) Serializable
B) Cloneable
C) Remote
D) Map
ANSWER: D

predict the problem in the below code

```
abstract class AbstractClass
{
   private abstract int abstractMethod();
}
```

A) Valid Syntax
B) No private access specifier for abstract method
C) Abstract keyword is not required for methods
D)declare abstract class with public
ANSWER: B

For all the Java classes the root class is
A) Event
B) Object
C) Throwable
D) Exception
ANSWER :B

Declaring an array with negative size throws
A)ArrayIndexOutofBoundsException
B)NegativeArraySizeException
C)Default value will be assinged
D)No compilation and runtime error
ANSWER:D

static methods are
A) Different for each object
B) Binded at compile time
C) Binded at runtime
D) similar to the instance methods
ANSWER:B

Which of the following is not a method of Character class
A)isDigit(char ch)
B)isWhiteSpace(char ch)
C)isLowerCase(char ch)
D)isNumber(char ch)
ANSWER:D