# 

# 

# 

# 

# LAB 02

# FLOW CONTROL STATEMENTS

EXERCISE 1

Write a Java program to input a number from user and print it into words using for loop. How to display number in words using loop in Java programming.

Logic to print number in words in Java programming.

Example

Input

1234  
Output  
One Two Three Four

Input:

16

Output:

one six

**For example:**

| **Test** | **Input** | **Result** |
| --- | --- | --- |
| 1 | 45 | Four Five |
| 2 | 13 | One Three |
| 3 | 87 | Eight Seven |

Answer:

import java.util.\*;

public class Words{

public static void main(String args[]){

Scanner S= new Scanner(System.in);

int a=S.nextInt(),d=0;

while(a!=0){

int b=a%10;

d=d\*10+b;

a/=10;

}

while(d!=0){

int b=d%10;

switch(b){

case 1:

System.out.print("One ");

break;

case 2:

System.out.print("Two ");

break;

case 3:

System.out.print("Three ");

break;

case 4:

System.out.print("Four ");

break;

case 5:

System.out.print("Five ");

break;

case 6:

System.out.print("Six ");

break;

case 7:

System.out.print("Seven ");

break;

case 8:

System.out.print("Eight ");

break;

case 9:

System.out.print("Nine ");

break;

default:

System.out.print("Zero ");

}

d/=10;}

}

}

|  | **Test** | **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- | --- |
|  | 1 | 45 | Four Five | Four Five |  |
|  | 2 | 13 | One Three | One Three |  |
|  | 3 | 87 | Eight Seven | Eight Seven |  |

EXERCISE 2

You and your friend are movie fans and want to predict if the movie is going to be a hit!

The movie’s success formula depends on 2 parameters:

the acting power of the actor (range 0 to 10)

the critic’s rating of the movie (range 0 to 10)

The movie is a hit if the acting power is excellent (more than 8) or the rating is excellent (more than 8). This holds true except if either the acting power is poor (less than 2) or rating is poor (less than 2), then the movie is a flop. Otherwise the movie is average.

Write a program that takes 2 integers:

the first integer is the acting power

second integer is the critic’s rating.

You have to print Yes if the movie is a hit, Maybe if the movie is average and No if the movie is flop.

Example input:

9 5

Output:

Yes

Example input:

1 9

Output:

No

Example input:

6 4

Output:

Maybe

**For example:**

| **Input** | **Result** |
| --- | --- |
| 9 5 | Yes |
| 1 9 | No |
| 6 4 | Maybe |

Answer:

import java.util.\*;

public class Movie{

public static void main(String args[]){

Scanner S=new Scanner(System.in);

int a=S.nextInt();

int b=S.nextInt();

if(a>8)

System.out.println("Yes");

else if(a<2||b<2)

System.out.println("No");

else

System.out.println("Maybe");

}

}

|  | **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | 9 5 | Yes | Yes |  |
|  | 1 9 | No | No |  |
|  | 6 4 | Maybe | Maybe |  |

EXERCISE 3

Write a program that takes as parameter an integer n.

You have to print the number of zeros at the end of the factorial of n.

For example, 3! = 6. The number of zeros are 0. 5! = 120. The number of zeros at the end are 1.

Note: n! < 10^5

Example Input:

3

Output:

0

Example Input:

60

Output:

14

Example Input:

100

Output:

24

Example Input:

1024

Output:

253

**For example:**

| **Input** | **Result** |
| --- | --- |
| 3 | 0 |
| 60 | 14 |
| 100 | 24 |
| 1024 | 253 |

Answer:

// Java program to count trailing 0s in n!

import java.io.\*;

import java.util.Scanner;

class prog {

// Function to return trailing

// 0s in factorial of n

static int findTrailingZeros(int n)

{

if (n < 0) // Negative Number Edge Case

return -1;

// Initialize result

int count=0;

// Keep dividing n by powers

// of 5 and update count

for (int i = 5; n / i >= 1; i\*=5 )

count += n / i;

return count;

}

// Driver Code

public static void main(String[] args)

{

int n ;

Scanner sc= new Scanner(System.in);

prog p=new prog();

System.out.println( p.findTrailingZeros(sc.nextInt()));

}

}

|  | **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | 3 | 0 | 0 |  |
|  | 60 | 14 | 14 |  |
|  | 100 | 24 | 24 |  |
|  | 1024 | 253 | 253 |  |