# LAB-02-LOGIC BUILDING

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 |

**CODE:**

```
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) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        System.out.println(findTrailingZeros(n));
    }
}
```

**OUTPUT:**

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

Passed all tests! ✓

Consider a sequence of the form 0, 1, 1, 2, 4, 7, 13, 24, 44, 81, 149...

Write a method program which takes as parameter an integer n and prints the nth term of the above sequence. The nth term will fit in an integer value.

Example Input:

5

Output:

4

Example Input:

8

Output:

24

Example Input:

11

Output:

149

**For example:**

| Input | Result |
|---|---|
| 5 | 4 |
| 8 | 24 |
| 11 | 149 |

**CODE:**

```
import java.util.Scanner;


public class SequenceFinder {


   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);


      int n = scanner.nextInt();


      int result = findNthTerm(n);
      System.out.println(result);
   }


   public static int findNthTerm(int n) {
      if (n <= 0) {
         return 0;
```

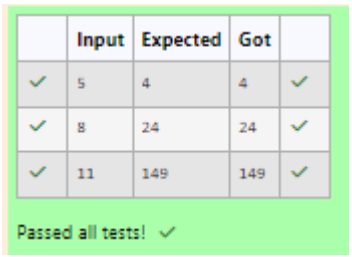
```
        }
        if (n == 1) {
            return 0;
        }
        if (n == 2 || n == 3) {
            return 1;
        }

        // Create an array to store the sequence terms
        int[] sequence = new int[n];
        sequence[0] = 0; // 1st term
        sequence[1] = 1; // 2nd term
        sequence[2] = 1; // 3rd term

        for (int i = 3; i < n; i++) {
            // The nth term is the sum of the previous terms
            sequence[i] = sequence[i - 1] + sequence[i - 2] + sequence[i - 3];
        }

        return sequence[n - 1];
    }
}
```

**OUTPUT:**



| | Input | Expected | Got | |
|---|---|---|---|---|
| ✓ | 5 | 4 | 4 | ✓ |
| ✓ | 8 | 24 | 24 | ✓ |
| ✓ | 11 | 149 | 149 | ✓ |

Passed all tests! ✓

You have recently seen a motivational sports movie and want to start exercising regularly. Your coach tells you that it is important to get up early in the morning to exercise. She sets up a schedule for you:
On weekdays (Monday - Friday), you have to get up at 5:00. On weekends (Saturday & Sunday), you can wake up at 6:00. However, if you are on vacation, then you can get up at 7:00 on weekdays and 9:00 on weekends.
Write a program to print the time you should get up.
Input Format
Input containing an integer and a boolean value.
The integer tells you the day it is (1-Sunday, 2-Monday, 3-Tuesday, 4-Wednesday, 5-Thursday, 6-Friday, 7-Saturday). The boolean is true if you are on vacation and false if you're not on vacation.
You have to print the time you should get up.
Example Input:
1 false
Output:
6:00
Example Input:
5 false
Output:
5:00
Example Input:
1 true
Output:
9:00

**For example:**

| Input | Result |
|---|---|
| 1 false | 6:00 |
| 5 false | 5:00 |
| 1 true | 9:00 |

**CODE:**

```
import java.util.Scanner;


public class WakeUpTime {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);


        int day = scanner.nextInt();
        boolean onVacation = scanner.nextBoolean();


        String wakeUpTime;


        if (onVacation) {
            if (day == 1 || day == 7) {
                wakeUpTime = "9:00"; // Saturday or Sunday
            } else {
                wakeUpTime = "7:00"; // Monday to Friday
            }
        } else {
```

```
        if (day == 1) {
            wakeUpTime = "6:00"; // Sunday
        } else if (day >= 2 && day <= 6) {
            wakeUpTime = "5:00"; // Monday to Friday
        } else {
            wakeUpTime = "6:00"; // Saturday
        }
    }

    System.out.println(wakeUpTime);
  }
}
```

**OUTPUT:**

| | Input | Expected | Got | |
|---|---|---|---|---|
| ✓ | 1 false | 6:00 | 6:00 | ✓ |
| ✓ | 5 false | 5:00 | 5:00 | ✓ |
| ✓ | 1 true | 9:00 | 9:00 | ✓ |

Passed all tests! ✓