Status	Finished
Started	Sunday, 22 September 2024, 10:58 AM
Completed	Sunday, 22 September 2024, 11:40 AM
Duration	41 mins 43 secs

Question **1**

Correct

Marked out of 5.00

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

Answer: (penalty regime: 0 %)

```
import java.util.Scanner;
public class CustomSequence
{
    public static int findterm(int n)
    {
        if(n==0) return 0;
        if(n==1||n==2) return 1;
        int[] sequence=new int[n+1];
        sequence[0]=0;
        sequence[1]=1;
        sequence[2]=1;
        for(int i=3;i<=n;i++)
        {
            sequence[i]=sequence[i-3]+sequence[i-2]+sequence[i-1];
        }
        return sequence[n-1];
    }
    public static void main(String[] args)
    {
        Scanner scanner=new Scanner(System.in);
        int n=scanner.nextInt();
        int nthTerm=findterm(n);
        System.out.println(nthTerm);
    }
}
```

	Input	Expected	Got	
✓	5	4	4	✓

	Input	Expected	Got	
✓	8	24	24	✓
✓	11	149	149	✓

Passed all tests! ✓

Question **2**

Correct

Marked out of 5.00

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

Answer: (penalty regime: 0 %)

```
import java.util.Scanner;
public class Timetable{
    public static void time(int day, boolean vacay){
        if(vacay == false){
            if(day == 1 || day == 7){
                System.out.println("6:00");
            }
            else{
                System.out.println("5:00");
            }
        }
        else{
            if(day == 1|| day == 7){
                System.out.println("9:00");
            }
            else{
                System.out.println("7:00");
            }
        }
    }
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);
        String input = sc.nextLine();
```

```
24          String[] parts = input.split(" ");
25          int day = Integer.parseInt(parts[0]);
26          boolean vacay = Boolean.parseBoolean(parts[1]);
27          time(day,vacay);
28
29      }
30
31  }
```

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

Passed all tests! ✓

Question **3**

Correct

Marked out of 5.00

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: (penalty regime: 0 %)

Reset 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)
    {

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

```

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

Passed all tests! ✓

◄ Lab-02-MCQ

Jump to...

Lab-03-MCQ ►