**Лабораторная работа 1. Простые типы данных**

**1. Вычислить количество цифр целого неотрицательного числа n.**

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Одно целое неотрицательное число n | Выведите количество цифр в числе n |
| 12343 | 5 |

import java.util.Scanner;

public class Lab1\_1 {

    public static void main(String[] args){

        Scanner sc = new Scanner(System.in);

        int number = sc.nextInt();

        sc.close();

        int count = 0;

        while(number > 0){

            number /= 10;

            count++;

         }

        System.out.println(count);

    }

}



**2. Найти сумму цифр целого числа.**

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Одно целое **32**-х разрядное число **n** (число может быть отрицательным). | Вывести сумму цифр числа **n**. |
| 321 | 6 |

import java.util.Scanner;

public class Lab1\_2 {

    public static void main(String[] args){

        Scanner sc = new Scanner(System.in);

        int number=sc.nextInt();

        sc.close();

        int sum = 0;

        while(number > 0){

            sum += number % 10;

            number /= 10;

        }

        System.out.println(sum);

    }

}



**3. Вычислите факториал числа.**

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Одно целое число **n** (**0** ≤ **n** ≤ **20**). | Выведите значение **n!** = **1** \* **2** \* **3** \* ... \* **n**. |
| 3 | 6 |

import java.util.Scanner;

public class Lab1\_3 {

    public static void main(String[] args){

        Scanner sc = new Scanner(System.in);

        int number = sc.nextInt();

        sc.close();

        int factorial = 1;

        for(int i = 2; i <= number; i++)

        {

            factorial \*= i;

        }

        System.out.println(factorial);

    }

}



**4. Дана функция, аргументы которой – неотрицательные целые числа m** **и** **n (m** ≤ **n):**

Вычислить значение функции.

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Два целых неотрицательных числа **n** и **m** (**0** ≤ **n**, **m** ≤ **20**). | Выведите искомое значение заданной функции **f**(**m**, **n**). |
| 4 2 | 6 |

import java.util.Scanner;

public class Lab1\_4 {

    public static int f(int m, int n)

    {

        if(m == 0 || m == n)

        {

            return 1;

        }

        return f(m-1,n-1) + f(m,n-1);

    }

    public static void main(String[] args){

        Scanner sc = new Scanner(System.in);

        int n = sc.nextInt();

        int m = sc.nextInt();

        sc.close();

        System.out.println(f(m,n));

    }

}



**5. Вычислить значение xn.**

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Два натуральных числа **x** и **n**. | Выведите значение **xn**, если известно что оно не превосходит **1018**. |
| 2 10 | 1024 |

import java.util.Scanner;

public class Lab1\_5 {

    public static void main(String[] args){

        Scanner sc = new Scanner(System.in);

        int x = sc.nextInt();

        int n = sc.nextInt();

        sc.close();

        int result = 1;

        for(int i = 0; i < n; i++)

        {

            result \*= x;

        }

        System.out.println(result);

    }

}



**6. Последовательность Фибоначчи задается следующим образом:**

* a0 = 0
* a1 = 1
* ak = ak-1 + ak-2

Для заданного **n** найти значение **n**-го элемента **an** последовательности Фибоначчи.

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Одно натуральное число **n** (**1** ≤ **n** ≤ **40**). | Вывести **n**-ый элемент последовательности Фибоначчи. |
| 2 | 1 |
| 5 | 5 |
| 8 | 21 |

import java.util.Scanner;

public class Lab1\_6 {

    public static int Fib(int n){

        return n <= 1 ? n : Fib(n-1) + Fib(n-2);

        }

    public static void main(String[] args){

        Scanner sc = new Scanner(System.in);

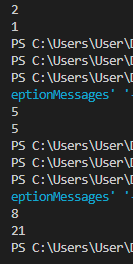
        int n = sc.nextInt();

        sc.close();

        System.out.println(Fib(n));

        }

}



**7. Собираясь как-то на экзамен, Петя разложил перед собой n разных шпаргалок со своего "любимого" предмета "Математический анализ". И, так как на протяжении семестра Петя не учился как следует, шпаргалок оказалось столько, что они все вместе не влазили ни в один карман. Тогда Петя посчитал максимальное количество шпаргалок, которое он может взять с собой на экзамен, и вдруг задумался, а сколько же всего существует способов выбрать нужное количество шпаргалок?**

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Содержит общее количество шпаргалок **n** (**1** ≤ **n** ≤ **12**) и количество шпаргалок **k** (**0** ≤ **k** ≤ **n**), которые Петя может взять с собой. | Вывести количество способов выбрать **k** шпаргалок из **n**. |
| 3 2 | 3 |
| 4 1 | 4 |

import java.util.Scanner;

public class Lab1\_7 {

    public static int Factorial(int n)

    {

        return n<= 0 ? 1 : n \* Factorial(n - 1);

    }

    public static void main(String[] args) {

        Scanner sc =new Scanner(System.in);

        int n = sc.nextInt();

        int k = sc.nextInt();

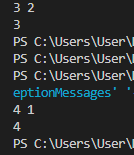
        sc.close();

         int result = Factorial(n) / (Factorial(n-k) \* Factorial(k));

        System.out.println(result);

    }

}



**8. Найти НОД (наибольший общий делитель) двух натуральных чисел.**

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Два натуральных числа **a** и **b** (**a**, **b** < **2** \* **109**). | Вывести НОД чисел **a** и **b**. |
| 42 24 | 6 |

import java.util.Scanner;

public class Lab1\_8 {

    public static int nod(int a, int b)

    {

        return (b == 0) ? a : nod(b, a % b);

    }

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        int a = sc.nextInt();

        int b = sc.nextInt();

        sc.close();

        System.out.println(nod(a,b));

    }

}

