



Министерство науки и высшего образования Российской Федерации  
Калужский филиал  
федерального государственного бюджетного  
образовательного учреждения  
высшего образования  
«Московский государственный технический университет имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(КФ МГТУ им. Н.Э. Баумана)

**ФАКУЛЬТЕТ ИУК "Информатика и управление"**

**КАФЕДРА ИУК5 "Системы обработки информации"**

## **ЛАБОРАТОРНАЯ РАБОТА №2**

**«РЕАЛИЗАЦИЯ ОСНОВНЫХ АЛГОРИТМИЧЕСКИХ КОНСТРУКЦИЙ»**

**ДИСЦИПЛИНА: «Основы программирования»**

Выполнил: студент гр.

\_\_\_\_\_  
(Подпись) \_\_\_\_\_ (Ф.И.О.)

\_\_\_\_\_  
(Подпись) \_\_\_\_\_ (Кондратьева С. Д.)  
(Ф.И.О.)

Проверила:

Дата сдачи (защиты):

Результаты сдачи (защиты):

- Балльная оценка:
- Оценка:

Калуга, 202

**Целью** лабораторной работы является формирование практических навыков по основам построения базовых алгоритмов

**Задачи:**

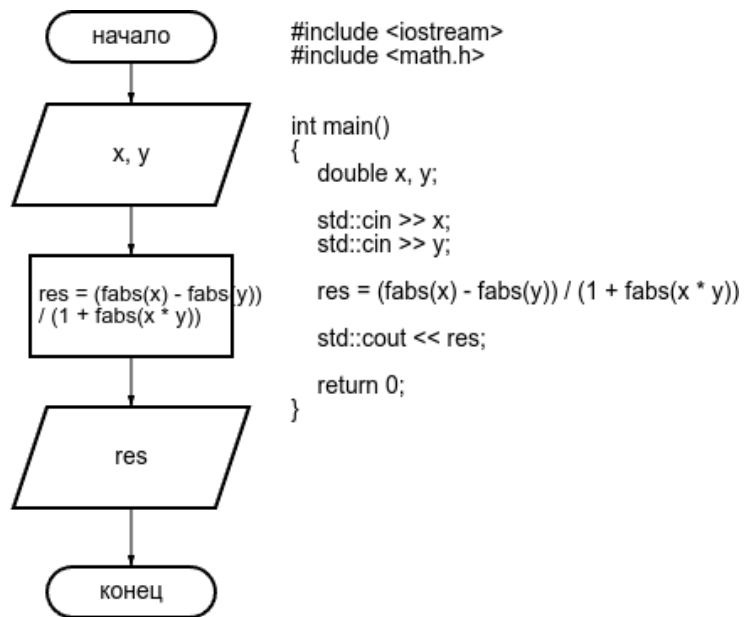
1. Овладеть навыками построения базовых алгоритмов.
2. Овладеть навыками выбора оптимальных алгоритмов программирования.
3. Овладеть навыками описания основных этапов построения алгоритмов
4. Овладеть навыками реализации основных алгоритмических конструкций

## Порядок выполнения

Задание 1:

Даны x,y. Получить:

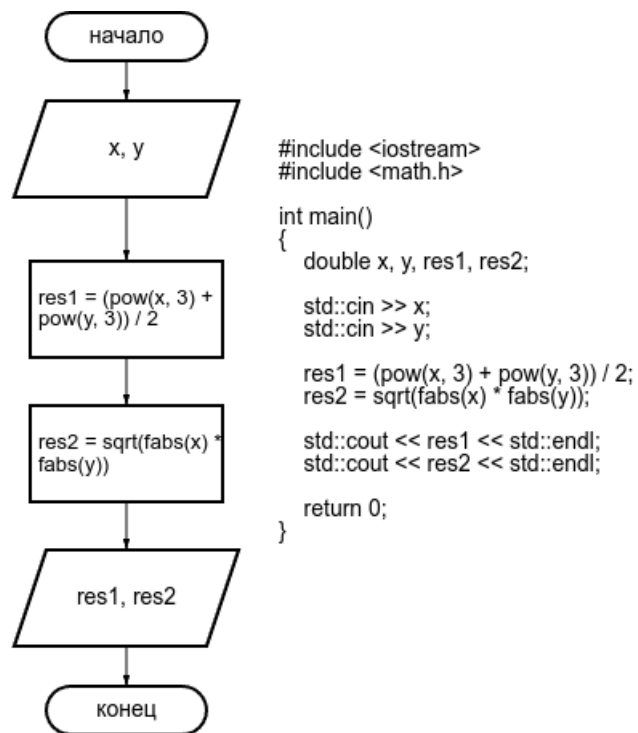
$$\frac{|x| - |y|}{1 + |xy|}.$$



```
10
5
0.0980392
```

## Задание 2:

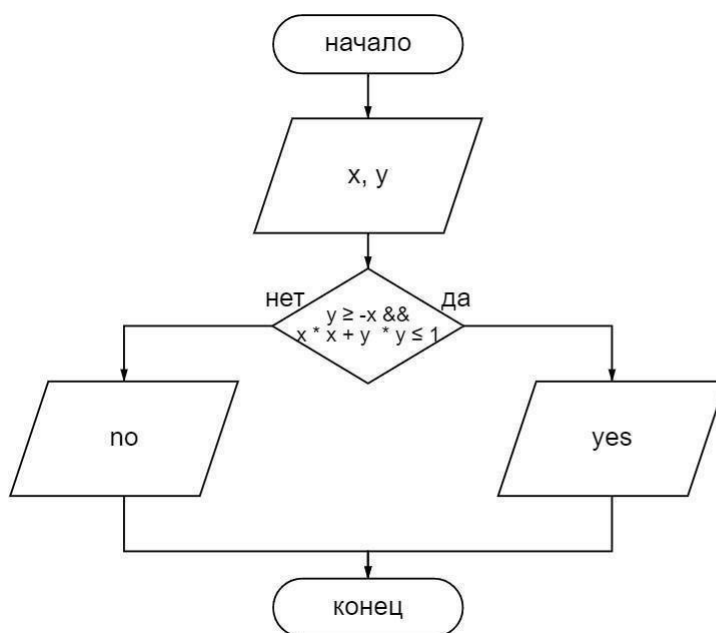
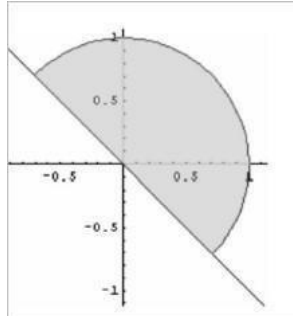
Даны 2 числа. Получить среднее арифметическое кубов этих чисел и среднее геометрическое модулей этих чисел.



```
10
5
562.5
7.07107
```

### Задание 3:

Определить, попадает ли заданная точка внутрь заданной области:



```
#include <iostream>
```

```
int main()
```

```
{  
    double x, y;
```

```
    std::cout << "Enter x: ";
```

```
    std::cin >> x;
```

```
    std::cout << "Enter y: ";
```

```
    std::cin >> y;
```

```
    if(y >= -x && x * x + y * y <= 1) {  
        std::cout << "yes";
```

```
    }
```

```
    else {
```

```
        std::cout << "no";
```

```
    }
```

```
    return 0;
```

```
}
```

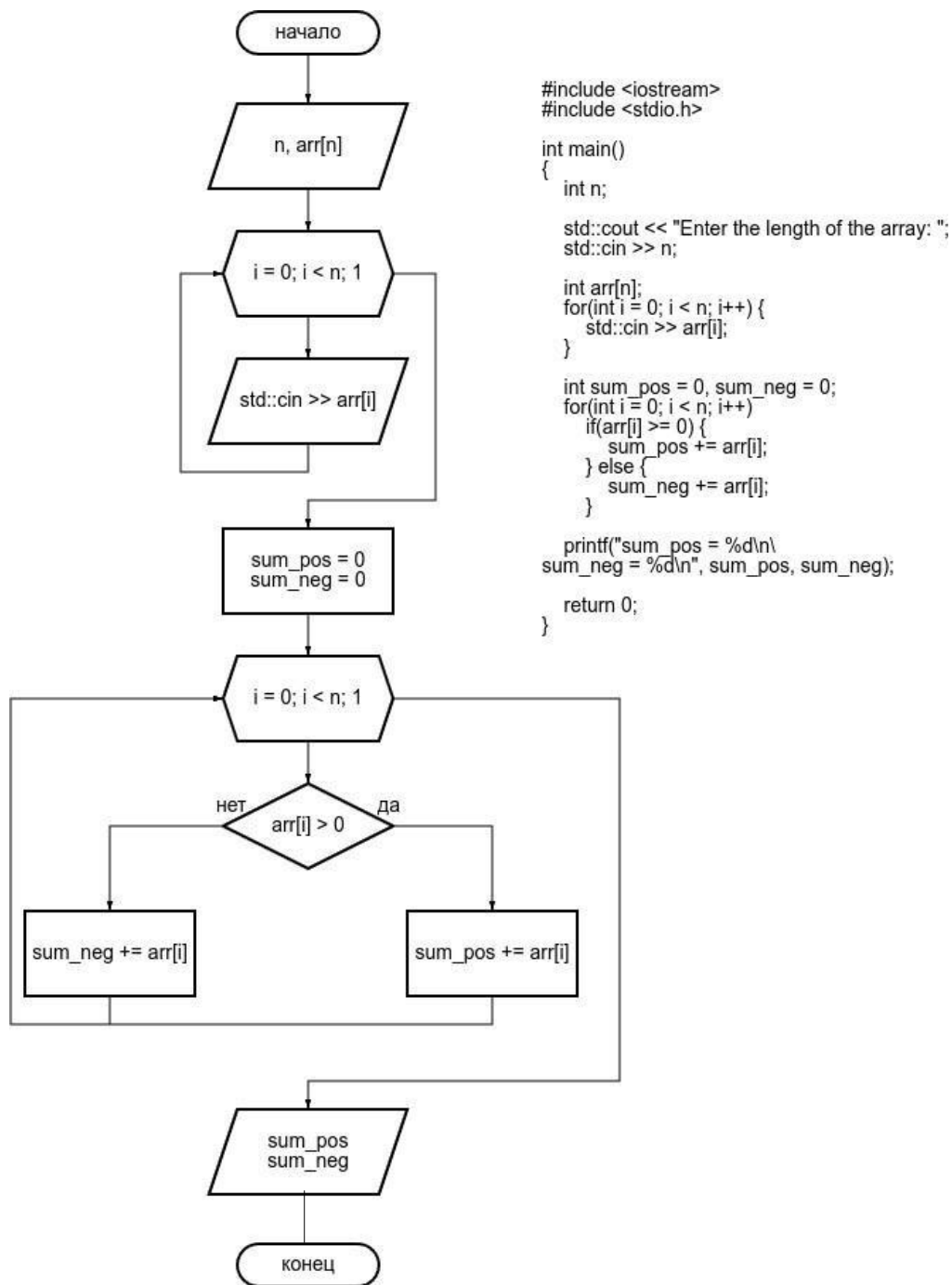
```
Enter x: 10
```

```
Enter y: 5
```

```
no
```

#### Задание 4:

Ввести одномерный массив из n элементов. Вычислить сумму всех отрицательных чисел и сумму всех положительных чисел.



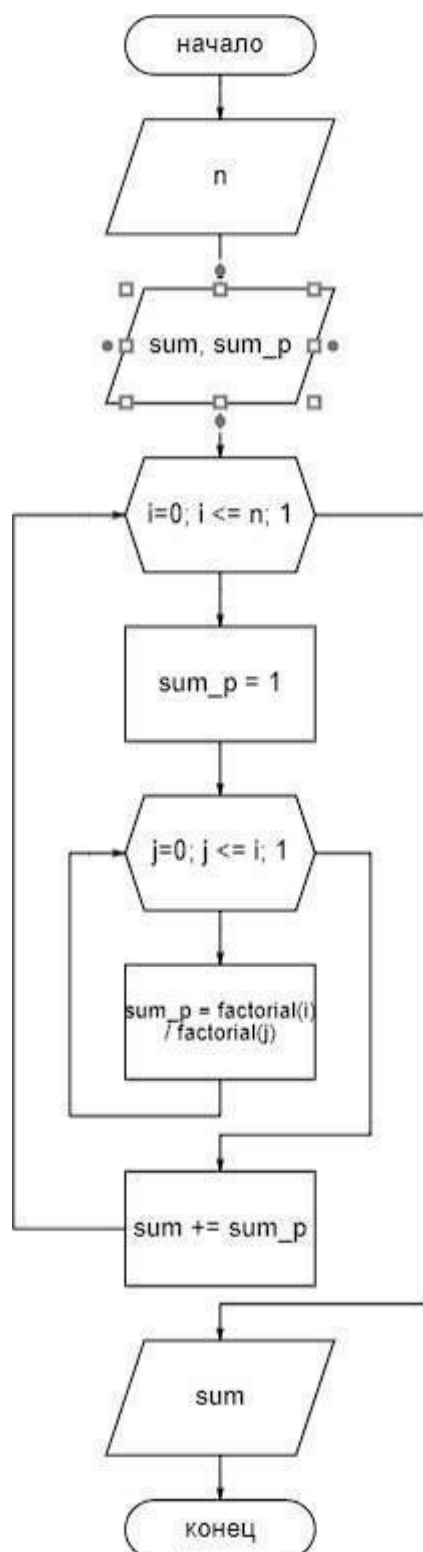
```

1
2
-2
-3
5
sum_pos = 8
sum_neg = -5
  
```

# Задание 5:

Дано натуральное число N. Вычислить:

$$S = \sum_{i=1}^N \prod_{j=1}^i \frac{j!}{i!}$$



```

#include <iostream>

int factorial(int num) {
    int fact = 1;
    for(; num; num--)
        fact *= num;
    return fact;
}

int main()
{
    int n;

    std::cout << "Enter n: ";
    std::cin >> n;

    long long int sum = 0, sum_p;
    for(int i = 0; i <= n; i++){
        sum_p = 1;
        for(int j = 0; j <= i; j++){
            sum_p *= factorial(i) / factorial(j);
        }
        sum += sum_p;
    }

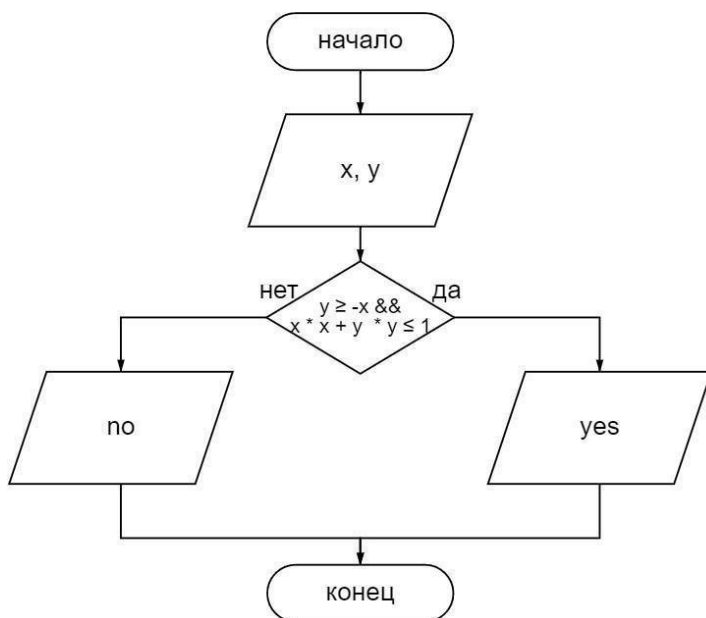
    std::cout << sum;

    return 0;
}
    
```

Enter n: 5  
86427762\_

## ОТВЕТЫ НА КОНТРОЛЬНЫЕ ВОПРОСЫ:

- 1) Схемы алгоритмов, программ, данных и систем (далее – схемы) состоят из имеющих заданное значение символов, краткого пояснительного текста и соединяющих линий.
- 2) Алгоритм — это чёткая последовательность действий, выполнение которой даёт какой-то заранее известный результат.
- 3) Различают три базовые алгоритмические структуры: следование, разветвление, цикл.
- 4) Условный оператор — это оператор, который реализует выполнение определённых команд при условии, что некоторое логическое выражение (условие) принимает значение «истина».
- 5) Выполнение операции или группы операций, в результате которых изменяется значение, форма представления или расположение данных
- 6) Выбор направления выполнения алгоритма или программы в зависимости от некоторых переменных условий.
- 7) Высота(a), ширина(b). a выбирается из ряда 10,15,20 мм.  $b = 1.5a$
- 8)



```
#include <iostream>

int main()
{
    double x, y;

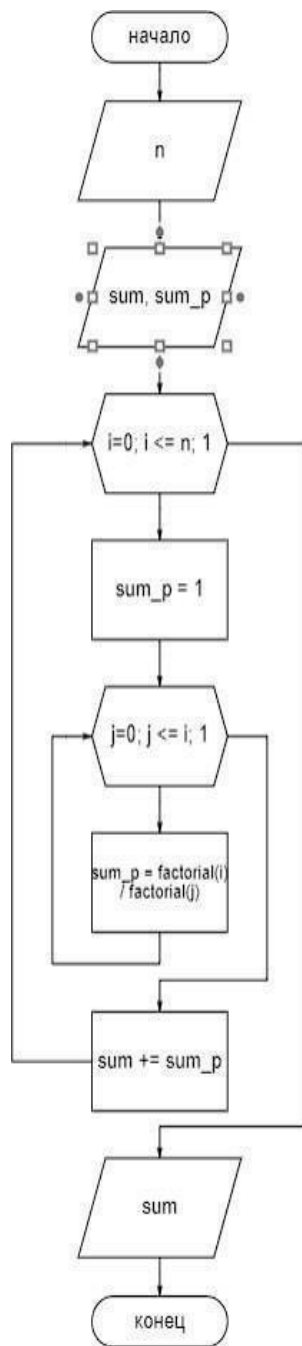
    std::cout << "Enter x: ";
    std::cin >> x;
    std::cout << "Enter y: ";
    std::cin >> y;

    if(y >= -x && x * x + y * y <= 1) {
        std::cout << "yes";
    }
    else {
        std::cout << "no";
    }

    return 0;
}
```



9)



```

#include <iostream>

int factorial(int num) {
    int fact = 1;
    for(; num; num--){
        fact *= num;
    }
    return fact;
}

int main()
{
    int n;

    std::cout << "Enter n: ";
    std::cin >> n;

    long long int sum = 0, sum_p;
    for(int i = 0; i <= n; i++){
        sum_p = 1;
        for(int j = 0; j <= i; j++){
            sum_p *= factorial(i) / factorial(j);
        }
        sum += sum_p;
    }

    std::cout << sum;

    return 0;
}
  
```

Enter n: 5  
86427762\_

**Вывод:** Были сформированы практические навыки по построению базовых алгоритмов