



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 ИУК4 «Программное обеспечение ЭВМ,  
информационные технологии»

## ДОМАШНЯЯ РАБОТА №2

### «Процедуры и функции»

ДИСЦИПЛИНА: «Высокоуровневое программирование»

Выполнил: студент гр. ИУК4-11Б

(подпись) ( Суриков Н.С. )  
(Ф.И.О.)

Проверил:

(подпись) ( Пчелинцева Н.И. )  
(Ф.И.О.)

Дата сдачи (защиты):

30.11.23

Результаты сдачи (защиты):

- Балльная оценка:

- Оценка:

70  
зачето

**Цель:** получить практические навыки по обработке одномерных массивов с помощью пользовательских подпрограмм на языке C++, научиться реализовывать рекурсивные функции.

**Задачи:**

1. Изучить виды подпрограмм: функции, возвращающие и не возвращающие значения.
2. Изучить механизм передачи параметров в подпрограммы по ссылке и значению.
3. Познакомиться с рекурсивными функциями.
4. Научиться создавать рекуррентные выражения.
5. Уметь определять глубину рекурсии и схематично представлять дерево вызовов рекурсии.
6. Закрепить полученные в ходе выполнения домашней работы навыки.

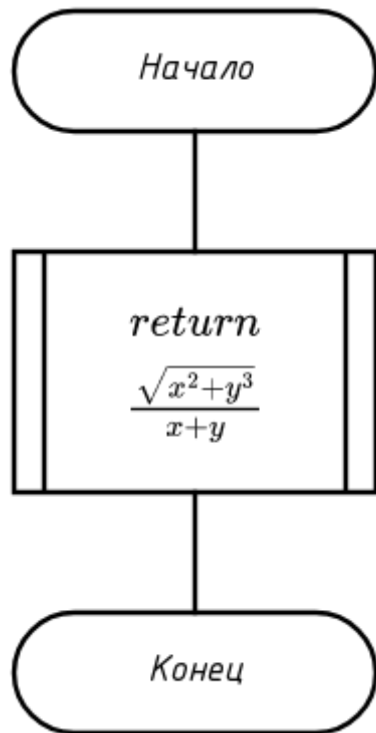
**Общее задание:**

1. Описать функцию для определения  $f(x,y)$  (вид функции для каждого варианта задания приведен в столбце 2 таблицы).
2. Задать конкретные значения  $s, t$  и вычислить  $a, b$  (формулы для определения  $a, b$  для каждого варианта задания приведены в столбце 2 таблицы).
3. Описать рекурсивные функции для определения  $p_i, h_i$ . Задать значение  $z$  и вычислить  $c$  (формулы для определения рекурсивных функций и для определения  $c$  приведены в столбце 3 таблицы).
4. Задать три последовательности случайных чисел  $in, jm, kl$  и произвести действия над каждым элементом этих последовательностей, используя процедуру (количество элементов последовательностей  $i, j, k$  и соответствующие действия для каждого варианта задания приведены в столбце 4 таблицы).
5. На экран необходимо последовательно вывести значения переменных  $a, b, c$ , заданные и преобразованные последовательности  $in, jm, kl$  (каждую последовательность чисел вывести в одну строку).

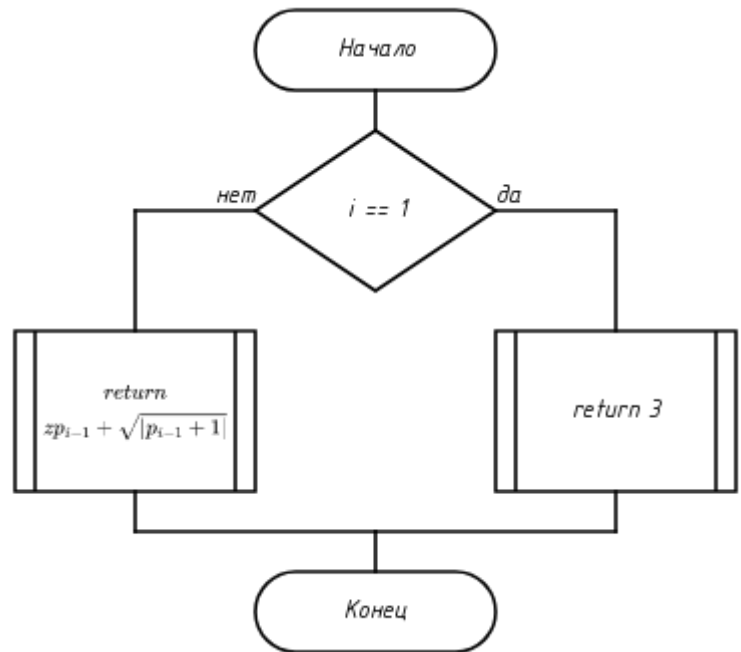
2	$f(x,y) = \frac{\sqrt{x^2 + y^3}}{x + y}$ $a = f(s,t) + f(s^2, t^2)$ $b = f(s, t^2)$	$p_1 = 3$ $h_1 = 8$ $p_i = zp_{i-1} + \sqrt{p_{i-1} + 1}$ $h_i = h_{i-1} / (2z) + 3h_{i-1}$ $c = (p_5 + h_3)^2$	$i = 8, j = 9, k = 10;$ все четные $n_i$ умножить на $a$ ; все четные $m_j$ умножить на $b$ ; все четные $l_k$ умножить на $c$ ;
---	--	---	--

## Вариант №2

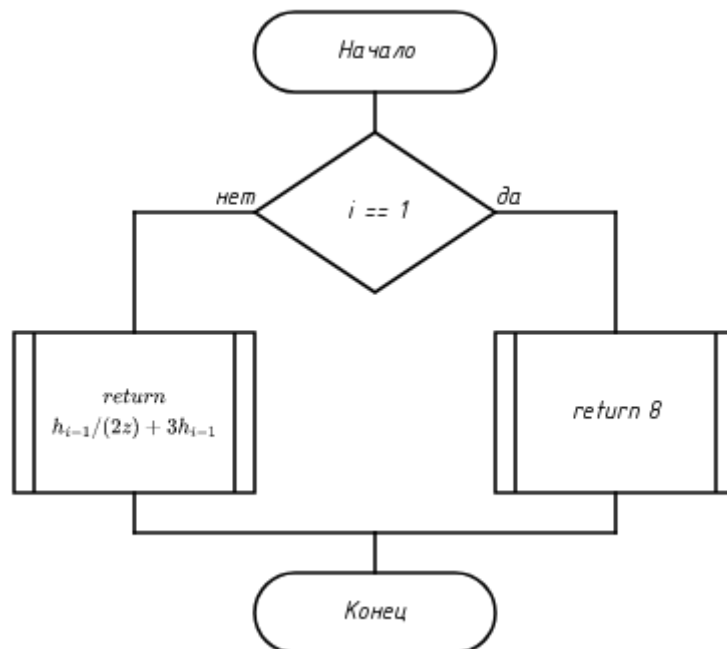
Блок схемы подпрограмм:



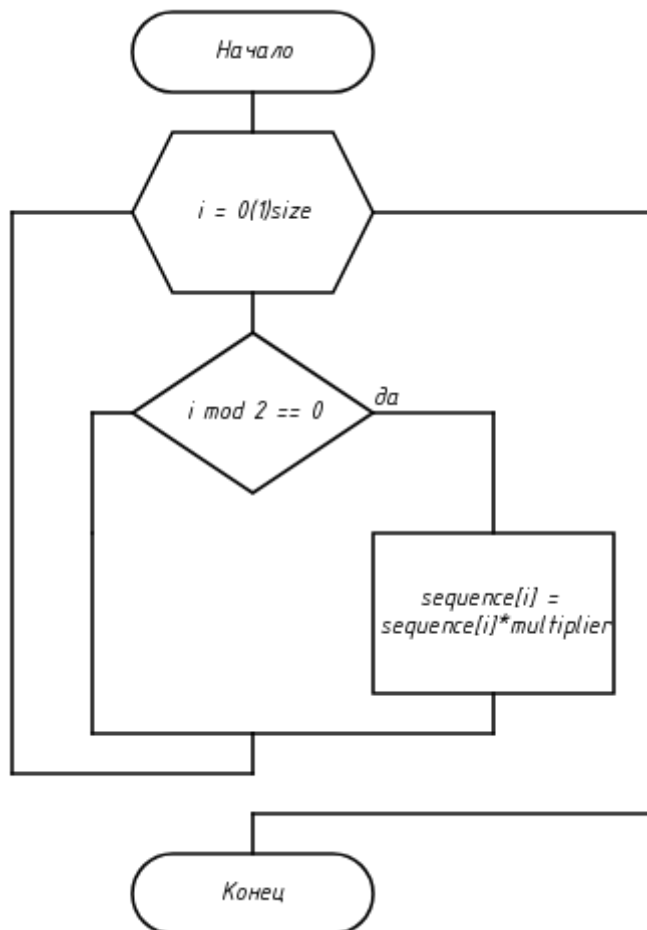
*double p(int i, double z)*



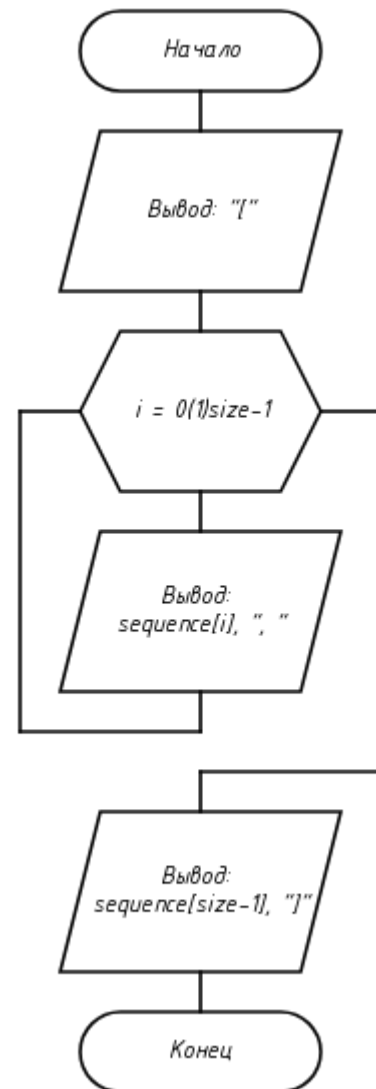
*double f(double x, double y)*



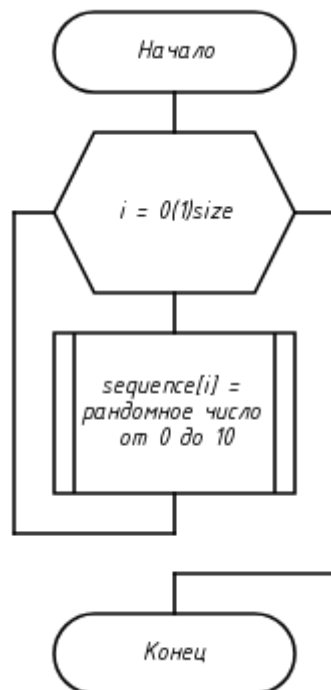
*double h(int i, double z)*



`void multEven(double *sequence, int size, double multiplier)`

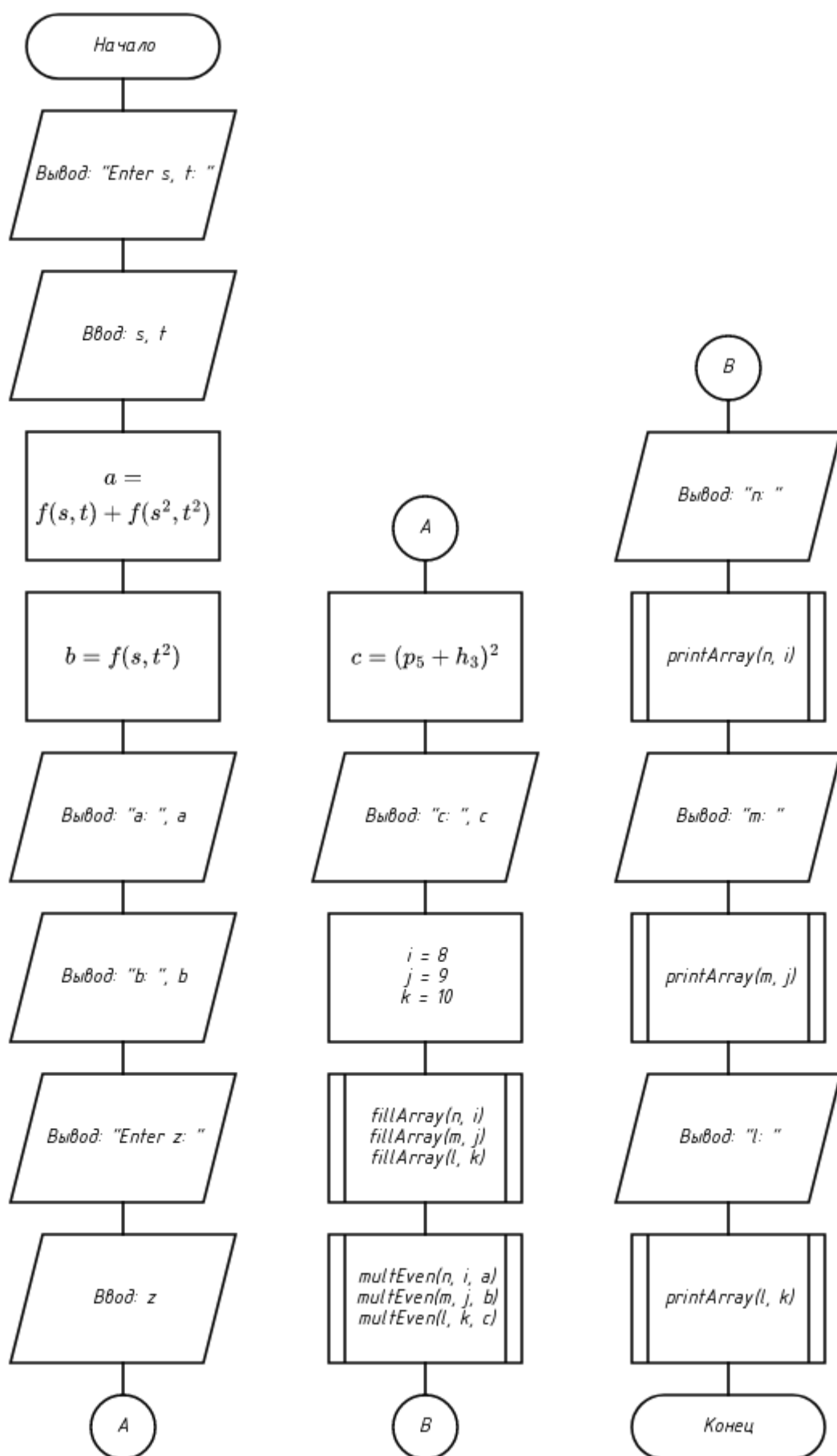


`void printArray(double *sequence, int size)`



`void fillArray(double *sequence, int size)`

Блок-схема основной программы:



*Листинг программы на C++:*

```
#include <iostream>
#include <cmath>
#include <cstdlib>
#include <ctime>

using namespace std;

double f(double x, double y)
{
    return sqrt(pow(x, 2) + pow(y, 3)) / x + y;
}

double p(int i, double z)
{
    if (i == 1)
        return 3;
    else
        return z * p(i - 1, z) + sqrt(fabs(p(i - 1, z) + 1));
}

double h(int i, double z)
{
    if (i == 1)
        return 8;
    else
        return h(i - 1, z) / (2 * z) + 3 * h(i - 1, z);
}

void multEven(double *sequence, int size, double multiplier)
{
    for (int i = 0; i < size; i++)
    {
        if (i % 2 == 0)
            sequence[i] *= multiplier;
    }
}

void printArray(double *sequence, int size)
{
    cout << "[";
    for (int i = 0; i < size - 1; i++)
    {
        cout << sequence[i] << ", ";
    }
    cout << sequence[size - 1] << "]" << std::endl;
}

void fillArray(double *sequence, int size)
{
    srand(time(nullptr));
```

```

    for (int i = 0; i < size; i++)
    {
        sequence[i] = 10.0 * rand() / (double)RAND_MAX;
    }
}

int main()
{
    double s{}, t{};
    cout << "Enter s, t: ";
    cin >> s >> t;
    double a = f(s, t) + f(pow(s, 2), pow(t, 2));
    double b = f(s, pow(t, 2));
    cout << "a: " << a << endl;
    cout << "b: " << b << endl;

    double z{};
    cout << "Enter z: ";
    cin >> z;
    double c = pow(p(5, z) + h(3, z), 2);
    cout << "c: " << c << endl;

    const int i = 8;
    const int j = 9;
    const int k = 10;

    double n[i]{};
    double m[j]{};
    double l[k]{};

    fillArray(n, i);
    fillArray(m, j);
    fillArray(l, k);

    multEven(n, i, a);
    multEven(m, j, b);
    multEven(l, k, c);

    std::cout << "n: ";
    printArray(n, i);

    std::cout << "m: ";
    printArray(m, j);

    std::cout << "l: ";
    printArray(l, k);
}

```

*Результат выполнения:*

*Enter s, t: 2 3*

*a: 21.6076*

*b: 22.537*

*Enter z: 1*

*c: 12482.7*

*n: [101.91, 2.12645, 191.782, 6.07269, 147.23, 2.13489, 197.226, 0.22591]*

*m: [106.294, 2.12645, 200.031, 6.07269, 153.563, 2.13489, 205.709, 0.22591, 25.6699]*

*l: [58873.7, 2.12645, 110793, 6.07269, 85055.3, 2.13489, 113938, 0.22591, 14218, 4.61457]*

*Enter s, t: 4 5*

*a: 40.8448*

*b: 56.266*

*Enter z: 2*

*c: 30989.2*

*n: [275.752, 3.22238, 233.601, 1.77216, 226.044, 0.605104, 163.807, 3.81591]*

*m: [379.864, 3.22238, 321.798, 1.77216, 311.389, 0.605104, 225.653, 3.81591, 340.374]*

*l: [209215, 3.22238, 177234, 1.77216, 171501, 0.605104, 124281, 3.81591, 187465, 6.04032]*

*Enter s, t: 6 7*

*a: 68.8248*

*b: 106.175*

*Enter z: 3*

*c: 191455*

*n: [529.78, 1.28489, 257.413, 8.24712, 325.565, 0.618975, 136.642, 5.9783]*

*m: [817.287, 1.28489, 397.109, 8.24712, 502.247, 0.618975, 210.797, 5.9783, 133.461]*

*l: [1.47373e+06, 1.28489, 716065, 8.24712, 905649, 0.618975, 380108, 5.9783, 240656, 1.22473]*

## **Вывод**

В ходе данной домашней работы я получил практические навыки по обработке одномерных массивов с помощью пользовательских подпрограмм на языке C++, научился реализовывать рекурсивные функции. Я изучил виды подпрограмм: функции, возвращающие и не возвращающие значения, механизм передачи параметров в подпрограммы по ссылке и значению, познакомился с рекурсивными функциями и научился создавать рекуррентные выражения.



## Литература

1. Курс лекций доцента кафедры ФН1-КФ Пчелинцевой Н.И.
2. Программирование на языке высокого уровня C/C++ [Электронный ресурс]: конспект лекций / – Электрон. текстовые данные. – М.: Московский государственный строительный университет, Ай Пи Эр Медиа, ЭБС АСВ, 2016. – 140 с. – Режим доступа: <http://www.iprbookshop.ru/48037>.
3. Зырянов, К. И. Программирование на C++: учебное пособие / К. И. Зырянов, Н. П. Кисленко. — Новосибирск: Новосибирский государственный архитектурно-строительный университет (Сибстрин), ЭБС АСВ, 2017. — 129 с. — ISBN 978-5-7795-0817- 9. — Текст: электронный // Электронно-библиотечная система IPR BOOKS: [сайт]. — URL: <https://www.iprbookshop.ru/85873.html>
4. Фридман, А. Л. Язык программирования C++: учебное пособие / А. Л. Фридман. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 217 с. — ISBN 978-5-4497-0920-2. — Текст: электронный // Электронно-библиотечная система IPR BOOKS: [сайт]. — URL: <https://www.iprbookshop.ru/102076.html>