

Национальный исследовательский ядерный университет «МИФИ»

Институт интеллектуальных кибернетических систем

Кафедра №12 «Компьютерные системы и технологии»



ОТЧЕТ

О выполнении лабораторной работы №2 «Вычисление значений числовых рядов и функций с заданной точностью»

Студент: Рыженко Р.В.

Группа: Б23-506

Преподаватель: Курочкина М-А.А.

Москва 2023

1. Формулировка индивидуального задания

Вариант №16. Вычислить значение функции в точке при помощи разложения в ряд.

2. Описание использованных типов данных

При выполнении данной лабораторной работы использовались встроенные типы данных `double` и `int`, предназначенные для работы с вещественными и целыми числами, и указатели, предназначенные для работы с адресами в памяти.

3. Описание использованного алгоритма

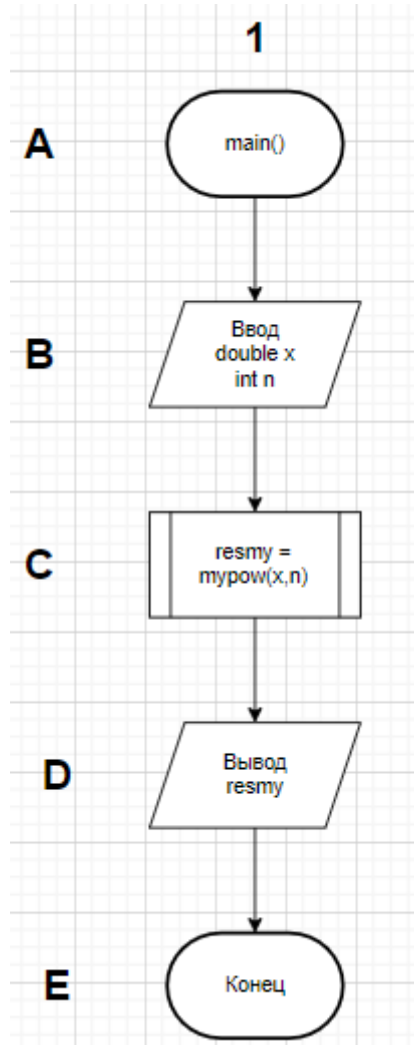


Рис. 1: Блок-схема алгоритма работы функции `main()` программы `prog1`

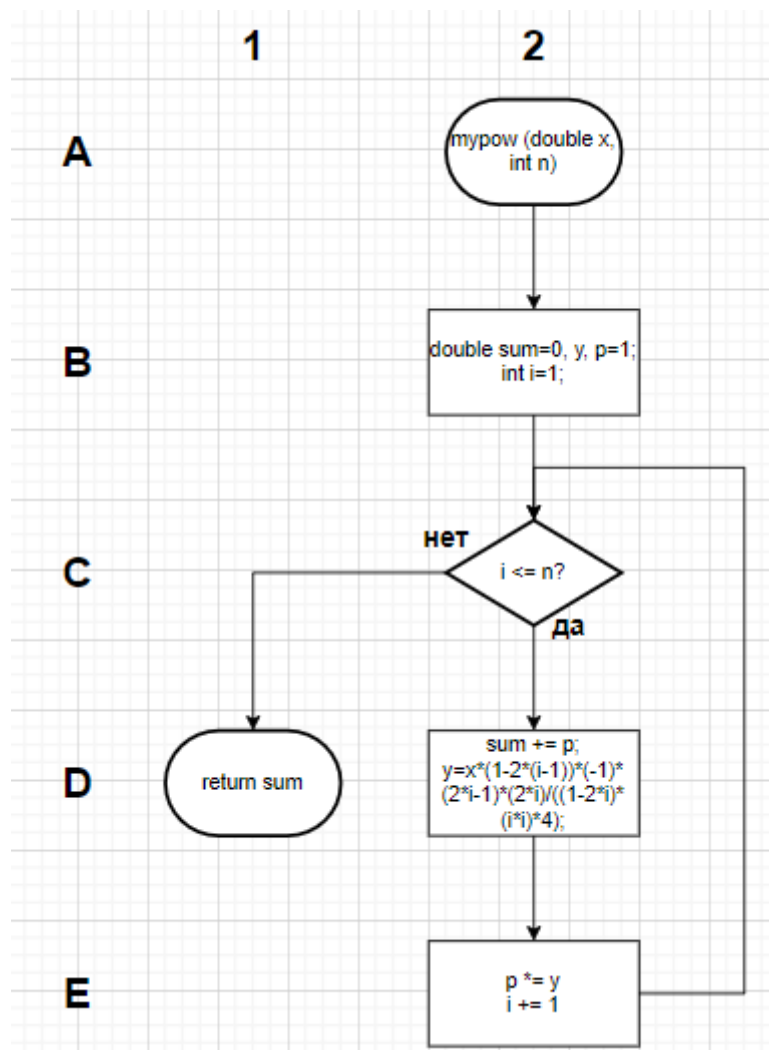


Рис. 2: Блок-схема алгоритма работы функции mypow () программы prog1

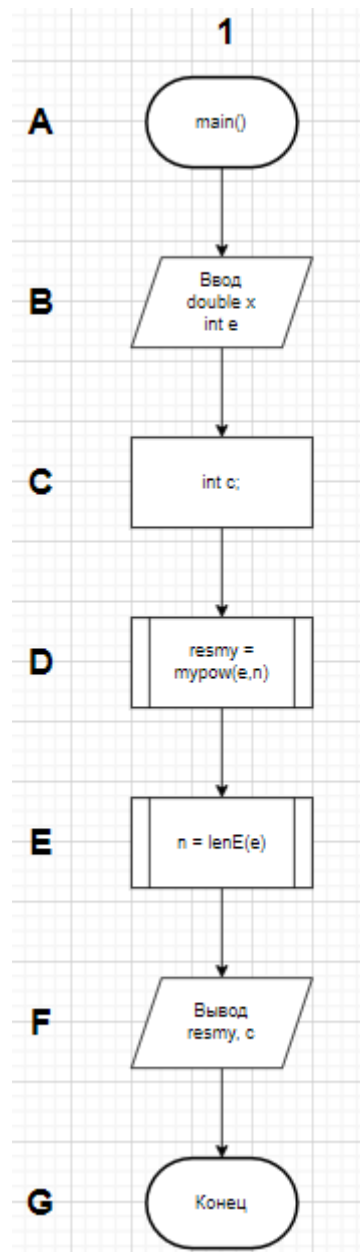


Рис. 3: Блок-схема алгоритма работы функции `main()` программы `prog2`

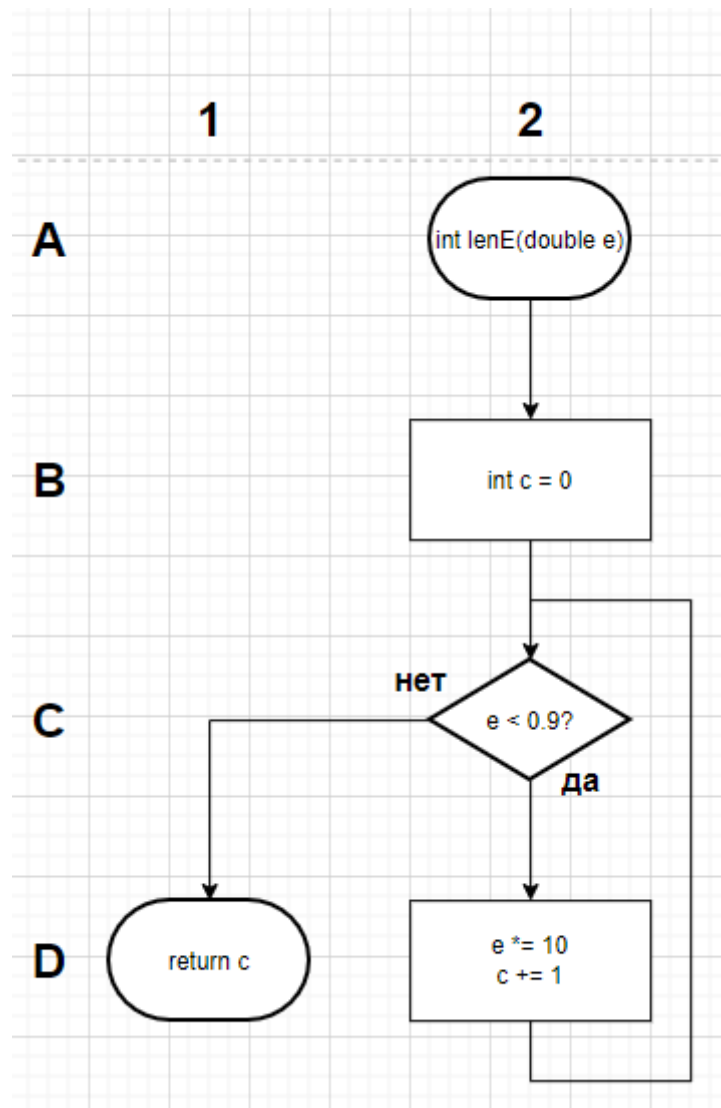


Рис. 4: Блок-схема алгоритма работы функции `lenE()` программы `prog2`

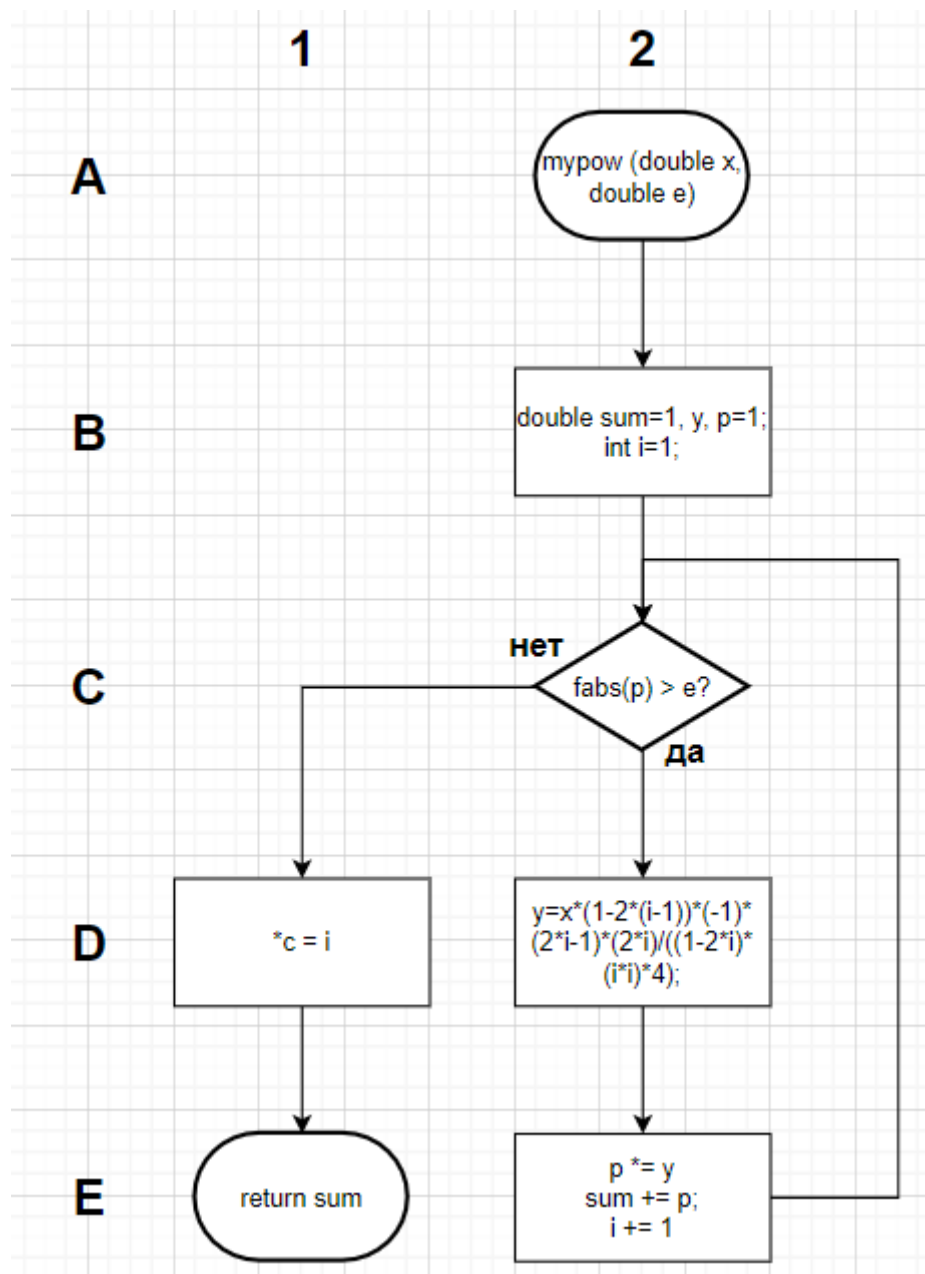


Рис. 5: Блок-схема алгоритма работы функции mypow () программы prog2

4. Исходные коды разработанных программ

Листинг 1: Исходные коды программы prog1 (файл: prog1.c)

```
#include <stdio.h>

double mypow(double x, int n) {
    double sum=0, y, p=1;
    int i=1;
    while (i <= n) {
        sum += p;
        y=x*(1-2*(i-1))*(-1)*(2*i-1)*(2*i)/((1-2*i)*(i*i)*4);
        p *= y;
        i++;
    }
    return sum;
}

int main()
{
    double x = -2;
    int n = -1;
    while (x <= -1 or x >= 1)
        while (scanf("%lf",&x) == 0)
            scanf("%*[^\\n]");
    while (n < 1)
        while (scanf("%d",&n) == 0)
            scanf("%*[^\\n]");

    double resmy = mypow(x,n);

    printf("%s%lf\\n", "sqrt(x+1) = ", resmy);

    return 0;
}
```

Листинг 2: Исходные коды программы prog2 (файл: prog2 . c)

```
#include <stdio.h>
#include <math.h>

double mypow(double x, double e, int *c) {
    int i=1;
    double sum=1, y, p=1;
    while (fabs(p) > e) {
        y=x*(1-2*(i-1))*(-1)*(2*i-1)*(2*i)/((1-2*i)*(i*i)*4);
        p *= y;
        sum += p;
        i++;
    }
    *c = i;
    return sum;
}

int lenE(double e) {
    int c=0;
    while (e < 0.9) {
        e*=10;
        c++;
    }
    return c;
}

int main()
{
    double x=-2, e;
    int c;
    while (x <= -1 or x >= 1)
        while (scanf("%lf",&x) == 0)
            scanf("%e",&x);
    while (scanf("%lf",&e) == 0)
        scanf("%e",&e);

    double resmy = mypow(x,e,&c);
    double resmath = pow(x+1,0.5);

    int n = lenE(e);
    printf("%s%.1f%s\n", "sqrt(x+1) = ", n+1, resmy, "\n(выведено на 1 цифру больше указанной точности, чтобы сравнить все нужные цифры без округления)");
    printf("%s%d\n", "Количество членов ряда: ", c);
    printf("%s%.1f%s\n", "Результат библиотеки math: ", n+1, resmath, "\n(выведено на 1 цифру больше указанной точности, чтобы сравнить все нужные цифры без округления)");

    return 0;
}
```

5. Описание тестовых примеров

Таблица 1: Тестовые примеры для программы 1

Значение x	Значение n	Ожидаемые значения	Полученные значения
0.5	3	1.218750	1.218750
0.5	30	1.224745	1.224745
0.2	1	1.000000	1.000000

Таблица 2: Тестовые примеры для программы 2

Значение x	Значение e	Ожидаемые значения	Полученные значения
0.5	0.1	1.2	1.2
0.5	0.0000001	1.2247448	1.2247448
0.2	1	1	1

6. Скриншоты

```
[ryzhenko.rv@unix lab2]$ ./prog1
0.5
3
1.218750
[ryzhenko.rv@unix lab2]$ ./prog1
0.5
30
1.224745
[ryzhenko.rv@unix lab2]$ ./prog1
0.2
1
1.000000
```

Рис. 6: Запуск программы prog1

```
[ryzhenko.rv@unix lab2]$ ./prog2
0.5
0.1
1.2
1.2
[ryzhenko.rv@unix lab2]$ ./prog2
0.5
0.0000001
1.2247448
1.2247448
[ryzhenko.rv@unix lab2]$ ./prog2
0.2
1
1
1
```

Рис. 7: Запуск программы prog2

7. Выводы

В ходе выполнения данной работы на примере программы, выполняющей вычисление значения функции в точке при помощи разложения в ряд, были рассмотрены базовые принципы работы построения программ на языке C и обработки целых чисел:

1. Организация ввода/вывода.
2. Разработка функций.
3. Объявление и использование переменных.
4. Выполнение простейших арифметических операций над целочисленными операндами.