

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: Баранов А.Т.

Группа: Б22-534

Преподаватель: Широких Т.А.

Москва — 2022

1. Формулировка индивидуального задания

Вариант №54

Задание

Вычислить значение функции в точке при помощи разложения в ряд:

$$\frac{1}{(1-x^2)\sqrt{1-x^2}} = 1 + \frac{3!!x^2}{2!!} + \frac{5!!x^4}{4!!} + \dots = \sum_{n=0}^{\infty} \frac{(2n+1)!!}{(2n)!!} x^{2n}$$

где $|x| < 1$.

2. Описание использованных типов данных

При выполнении данной лабораторной работы использовался встроенный тип данных `int`, предназначенный для работы с целыми числами, и встроенный тип данных `long double`, предназначенный для работы с числами с плавающей точкой повышенной точности.

3. Описание использованного алгоритма

Алгоритм 1: вычисления значения функции при возможности указать количество членов ряда, которое необходимо использовать при вычислениях.

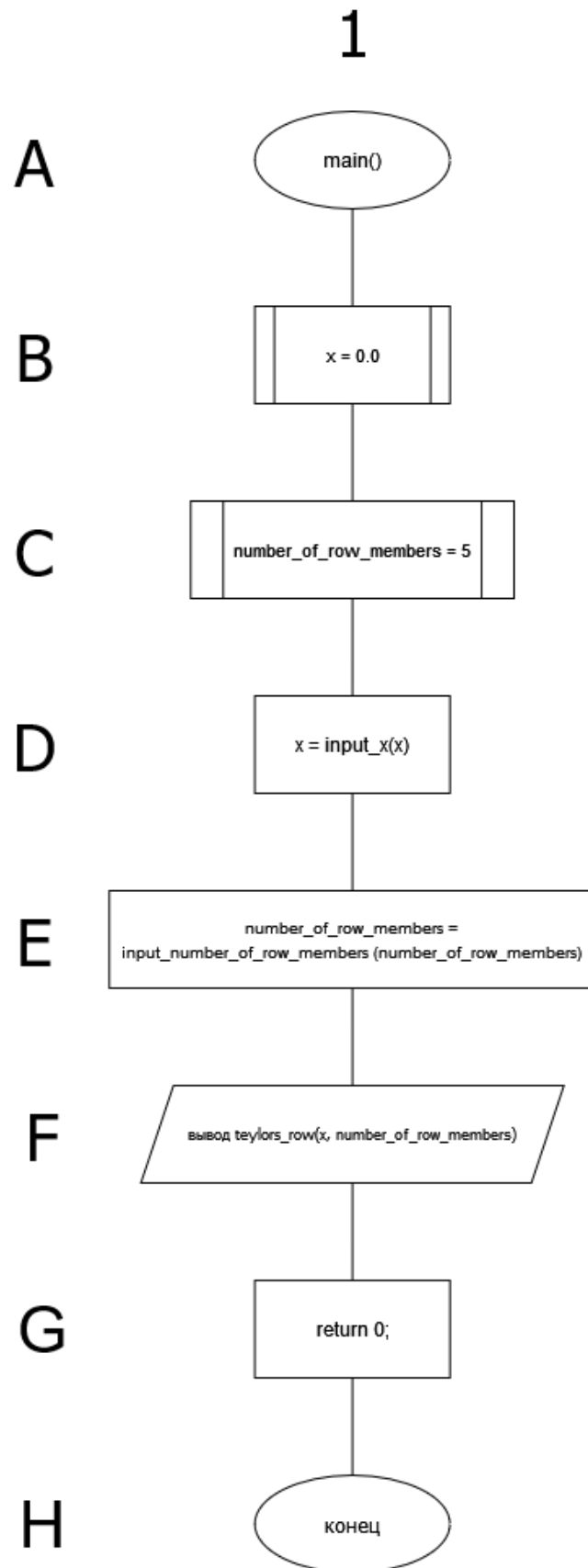


Рис. 1: Блок-схема алгоритма работы функции `main()`

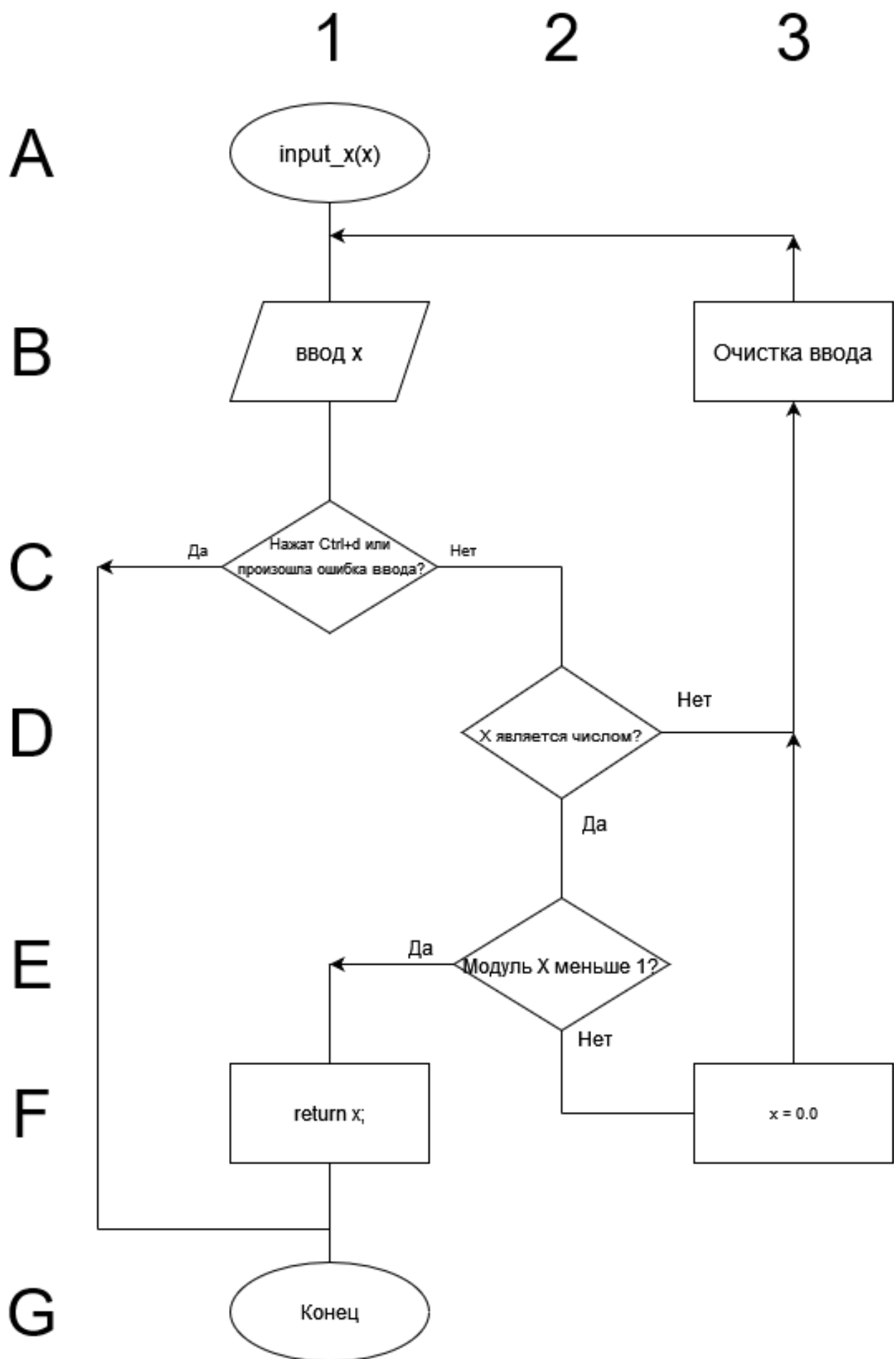


Рис. 2: Блок-схема алгоритма работы функции input_x()

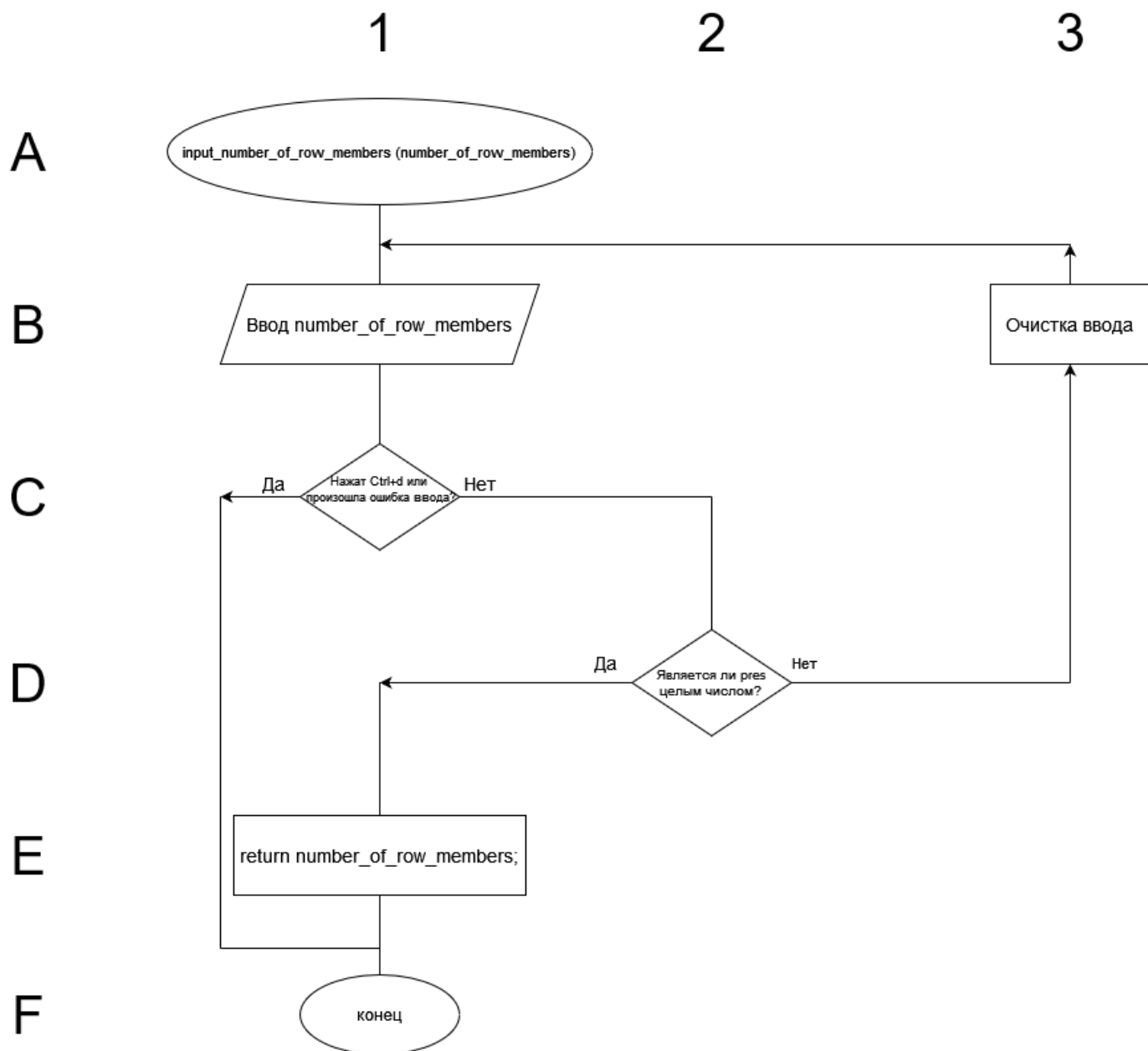


Рис. 3: Блок-схема алгоритма работы функции `input_number_of_row_members ()`

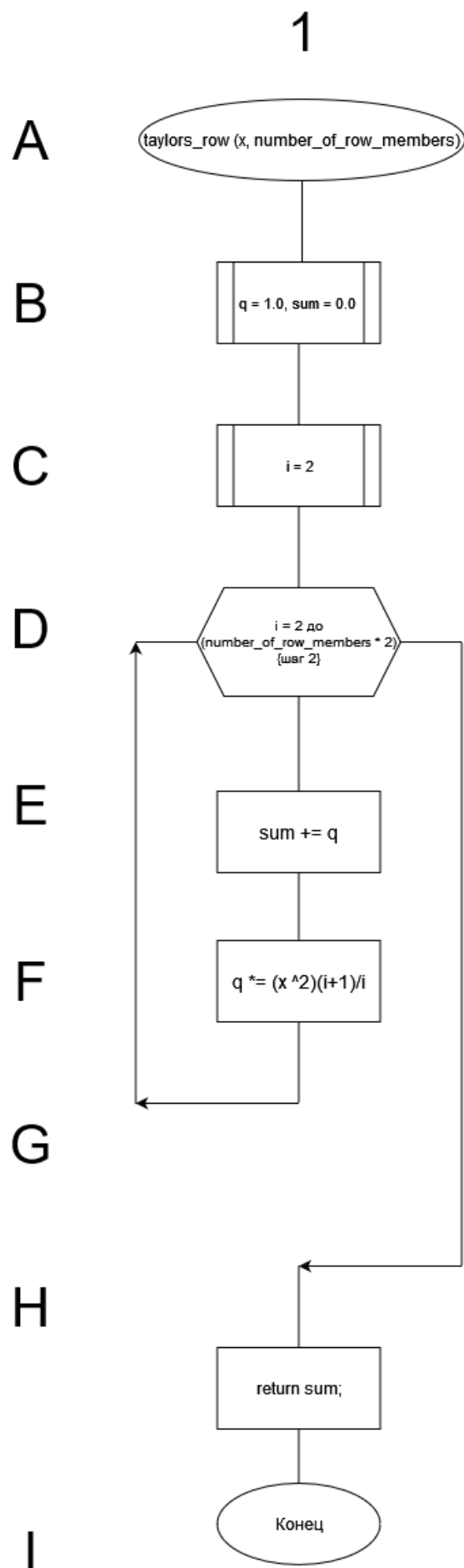


Рис. 4: Блок-схема алгоритма работы функции `taylors_row()`

Алгоритм 2: вычисления значения функции при возможности указать точность, с которой должно быть вычислено значение функции.

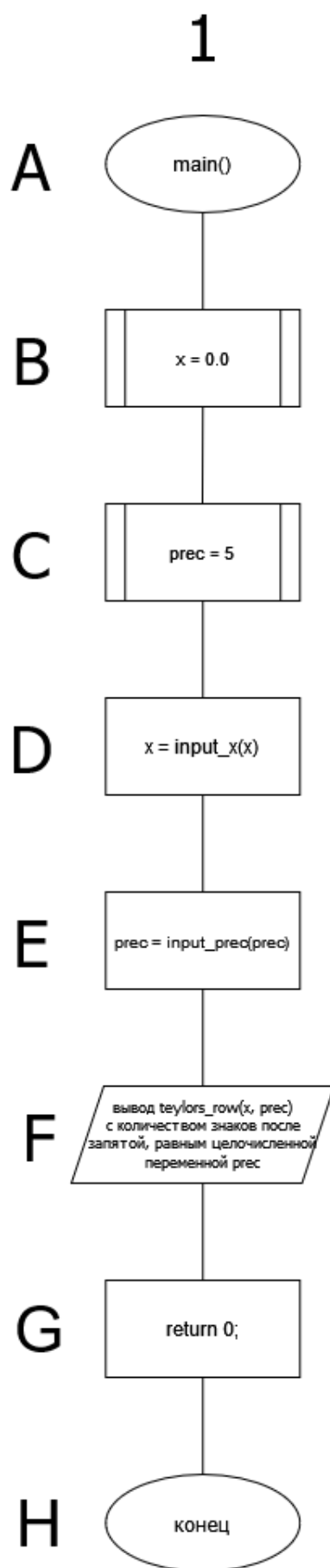


Рис. 5: Блок-схема алгоритма работы функции `main()`

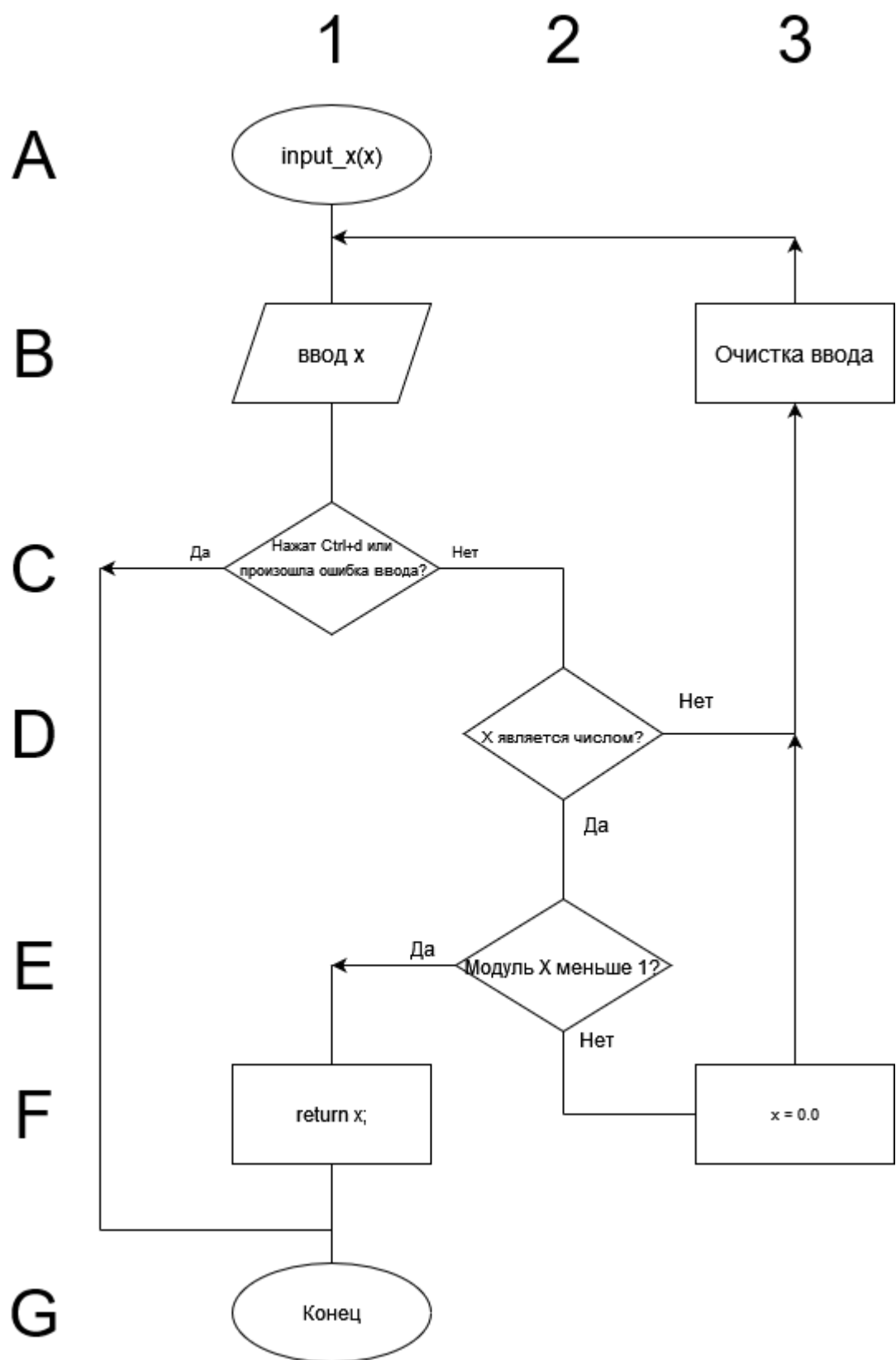


Рис. 6: Блок-схема алгоритма работы функции `input_x()`

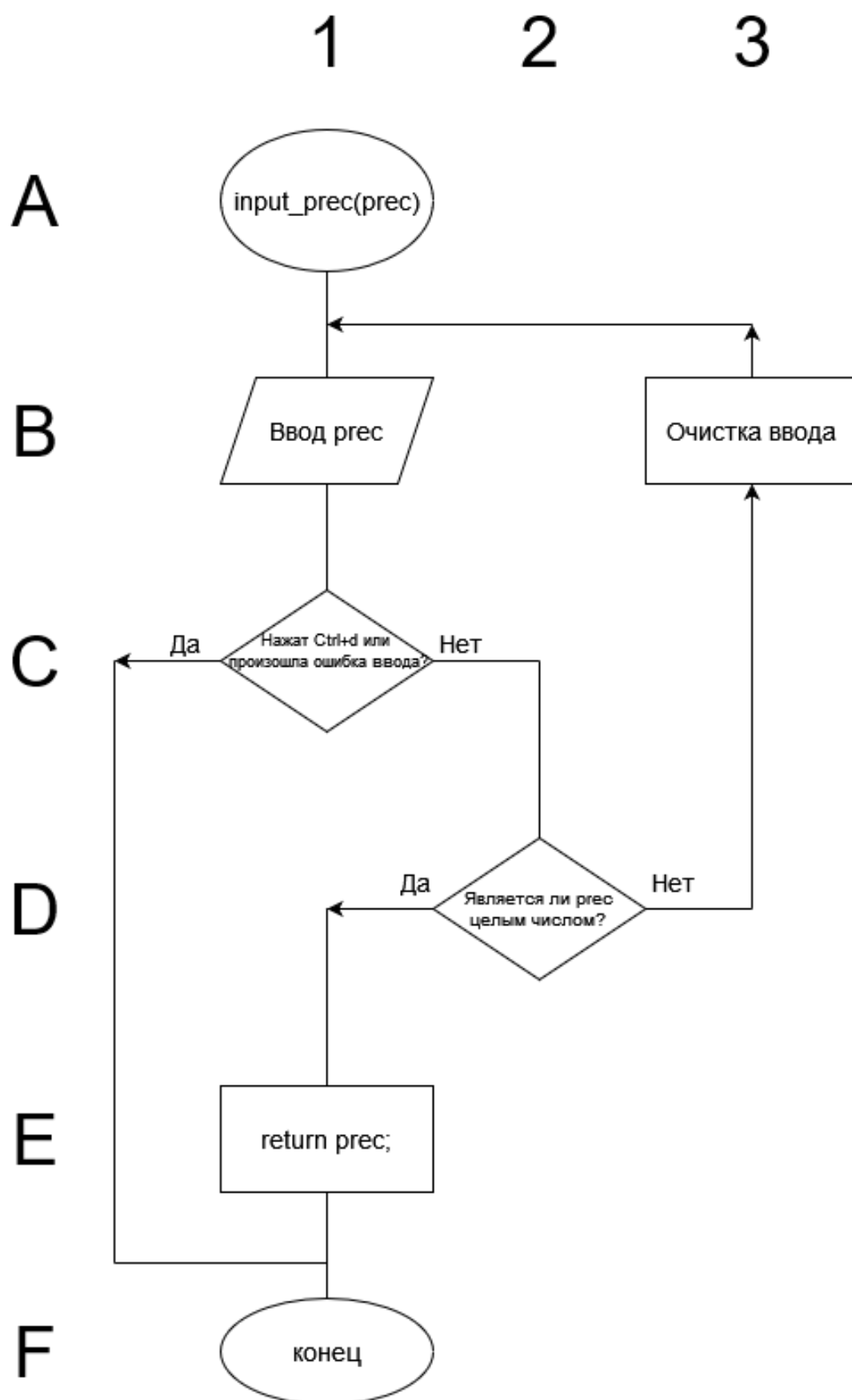


Рис. 7: Блок-схема алгоритма работы функции `input_prec()`

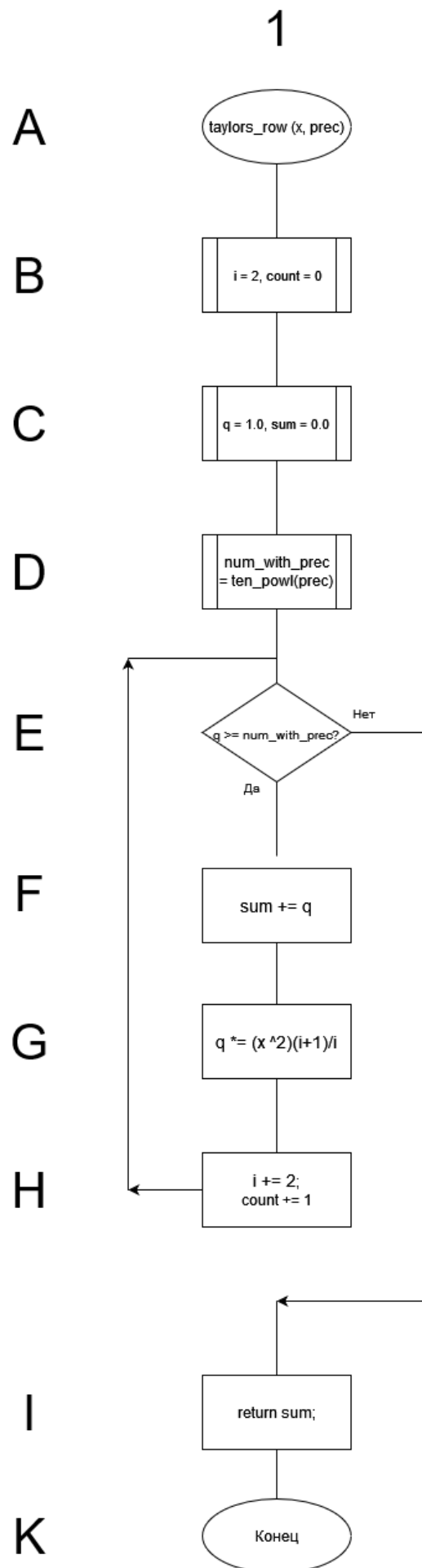


Рис. 8: Блок-схема алгоритма работы функции `taylors_row()`

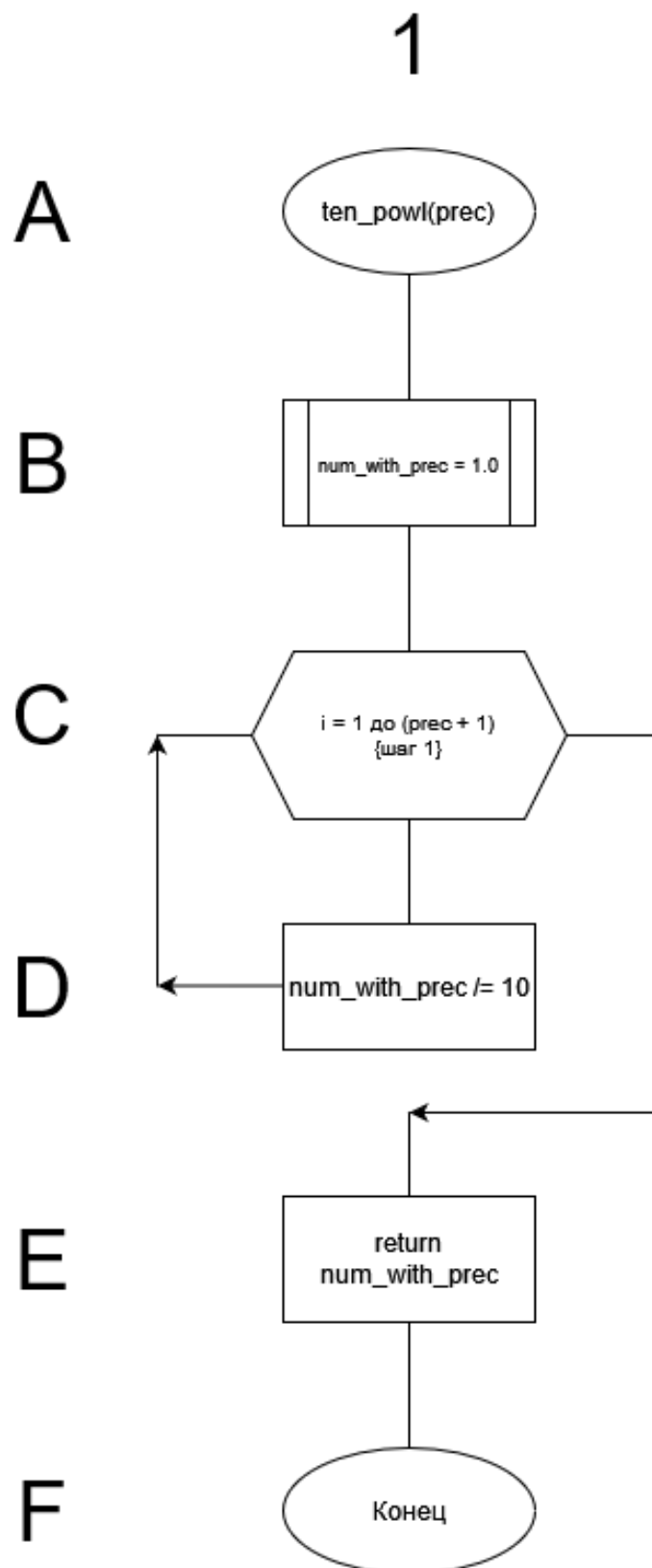


Рис. 9: Блок-схема алгоритма работы функции `ten_powl()`

4. Исходные коды разработанных программ

Листинг 1: Исходные коды программы code1 (файл: code1.c)

```
#include <stdio.h>
#include <iso646.h>
#include <math.h>

long double input_x(long double);
int input_number_of_row_members(int);
long double teylors_row(long double, int);

int main()
{
    long double x = 0.0;
    int number_of_row_members = 10;

    x = input_x(x);
    number_of_row_members = input_number_of_row_members(number_of_row_members);

    printf("\nCalculated value - %.25Lf\n", teylors_row(x, number_of_row_members));

    return 0;
}

long double input_x (long double x)
{
    printf("\nWrite number x in the range from -1 to 1 (except -1 and 1):");
    while (scanf("%Lf", &x) != 1 or fabs1(x) >= 1)
    {
        if (fabs1(x) >= 1)
        {
            printf("\nYou haven't written number in the range from -1 to 1"
                    "(except -1 and 1).Try again:");
            x = 0.0;
        }
        else //as for "if (scanf("%Lf", &x) != 1)"
        {
            if (getchar() == -1)
                printf("\nError or EOF\n");
            else
                printf("\nYou haven't written number.Try again:");
        }
        while (getchar() != '\n');
    }

    return x;
}

int input_number_of_row_members(int number_of_row_members)
{
    char check;

    printf("\nWrite integer required number of row members:");
    while ((scanf("%d%c", &number_of_row_members, &check) != 2) or (check != 10))
    {
        if (getchar() == -1)
            printf("\nError or EOF.\n");
        else
            printf("\nYou haven't written integer. Try again:");

        while (getchar() != '\n');
    }

    return number_of_row_members;
}
```

```

long double teylors_row(long double x, int number_of_members)
{
    long double q = 1.0, sum = 0.0;

    for(int i = 2; i/2 <= number_of_members; i += 2)
    {
        sum += q;
        q *= (x*x) * (i+1) / (i);
    }

    return sum;
}

```

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

```

#include <stdio.h>
#include <iso646.h>
#include <math.h>

long double input_x(long double);
int input_prec(int);
long double teylors_row(long double, int);
long double ten_pow1(int);

int main()
{
    long double x = 0.0;
    int prec = 5;

    x = input_x(x);
    prec = input_prec(prec);

    printf("\nCalculated value - %.Lf\n", prec, teylors_row(x, prec));

    return 0;
}

long double input_x (long double x)
{
    printf("\nWrite number x in the range from -1 to 1 (except -1 and 1):");
    while(scanf("%Lf", &x) != 1 or fabs1(x) >= 1)
    {
        if (fabs1(x) >= 1)
        {
            printf("\nYou haven't written number in range from -1 to 1"
                "(except -1 and 1). Try again:");
            x = 0.0;
        }
        else if (getchar() == -1)
            printf("\n\nError or EOF\n");
        else
            printf("\nYou haven't written number. Try again:");

        while(getchar() != '\n');
    }

    return x;
}

```

```

int input_prec(int prec)
{
    char check;

    printf("\nWrite integer precision (number of decimal places):");
    while ((scanf("%d%c", &prec, &check) != 2) or (check != 10))
    {
        if (check == -1)
            printf("\n\nError or EOF.\n");
        else
            printf("\nYou haven't written integer. Try again:");

        while(getchar() != '\n');
    }

    return prec;
}

long double teylors_row(long double x, int prec)
{
    int i = 2, count = 0;
    long double q = 1.0, num_with_prec = ten_powl(prec), sum = 0.0;

    while (q >= num_with_prec)
    {
        sum += q;
        q *= (x*x)*(i+1)/(i);
        i += 2;
        count += 1;
    }

    printf("\nNumber of row members - %d\n", count);

    return sum;
}

long double ten_powl(int prec) /*Example: pres = 3 -> num_with_pres = 0.0001 to better
*                               precision*/
{
    long double num_with_prec = 1;
    for (int i = 1; i <= (prec + 1); i++)
        num_with_prec /= 10;

    return num_with_prec;
}

```

5. Описание тестовых примеров

Таблица 1: Тестовые примеры программы №1.

Значение x	Значение number_of_row_members	Ожидаемое значение sum	Полученное значение sum
0.8	25	4.629395089770481181714380	4.629395089770481181714380
0.312879	30	1.1671135597614613957362373	1.1671135597614613957362373
0	500	1.000000000000000000000000000000	1.000000000000000000000000000000

6. Скриншоты

```
[baranov.at@unix:~/2nd_lab_work]$ ./code1.o

Write number x in the range from -1 to 1 (except -1 and 1):abc
You haven't written number. Try again:abc
You haven't written number. Try again:524.2
You haven't written number in the range from -1 to 1 (except -1 and 1). Try again:1
You haven't written number in the range from -1 to 1 (except -1 and 1). Try again:-1
You haven't written number in the range from -1 to 1 (except -1 and 1). Try again:0.8

Write integer required number of row members:abs
You haven't written integer. Try again:fjl
You haven't written integer. Try again:52.2
You haven't written integer. Try again:8541.2
You haven't written integer. Try again:25

Calculated value - 4.6293950897720481181714380

[baranov.at@unix:~/2nd_lab_work]$ ./code1.o

Write number x in the range from -1 to 1 (except -1 and 1):0.312879

Write integer required number of row members:30

Calculated value - 1.1671135597614613957362373

[baranov.at@unix:~/2nd_lab_work]$ ./code1.o

Write number x in the range from -1 to 1 (except -1 and 1):0

Write integer required number of row members:500

Calculated value - 1.000000000000000000000000000000
```

Рис. 10: Запуск программы prog1

Таблица 2: Тестовые примеры программы №2.

Значение x	Значение pres	Ожидаемое значение sum	Полученное значение sum
0.8	3	4.629	4.629
0.8	4	4.6296	4.6296
0	25	1.000000000000000000000000	1.000000000000000000000000
0.312879	6	1.167114	1.167114
-0.8	5	4.62963	4.62963

```
[baranov.at@unix:~/2nd_lab_work]$ ./code.o
Write number x in the range from -1 to 1 (except -1 and 1):0.8

Write integer presicion (number of decimal places):3

Number of row members - 25

Calculated value - 4.629

[baranov.at@unix:~/2nd_lab_work]$ ./code.o
Write number x in the range from -1 to 1 (except -1 and 1):0.8

Write integer presicion (number of decimal places):4

Number of row members - 30

Calculated value - 4.6296

[baranov.at@unix:~/2nd_lab_work]$ ./code.o
Write number x in the range from -1 to 1 (except -1 and 1):0

Write integer presicion (number of decimal places):25

Number of row members - 1

Calculated value - 1.000000000000000000000000

[baranov.at@unix:~/2nd_lab_work]$ ./code.o
Write number x in the range from -1 to 1 (except -1 and 1):0.312879

Write integer presicion (number of decimal places):6

Number of row members - 8

Calculated value - 1.167114

[baranov.at@unix:~/2nd_lab_work]$ ./code.o
Write number x in the range from -1 to 1 (except -1 and 1):abc

You haven't written number.Try again:52.2

You haven't written number in range from -1 to 1 (except -1 and 1).Try again:0.8

Write integer presicion (number of decimal places):as

You haven't written integer. Try again:53.2

You haven't written integer. Try again:2

Number of row members - 20

Calculated value - 4.63

[baranov.at@unix:~/2nd_lab_work]$ ./code.o
Write number x in the range from -1 to 1 (except -1 and 1):-0.8

Write integer presicion (number of decimal places):5

Number of row members - 36

Calculated value - 4.62963
```

Рис. 11: Запуск программы prog

7. Выводы

В ходе выполнения данной работы на примере программ, позволяющих вычислять значения некоторой заданной функции, были рассмотрены принципы работы построения программ на языке С и обработки чисел с плавающей точкой:

1. Организация ввода.
2. Разработка функций.
3. Объявление и использование переменных.
4. Вычисление значения заданной функции с помощью разложения её в ряд.
5. Работа с циклами и условными операторами.
6. Организация вывода с количеством знаков после запятой, введённым пользователем.