Национальный исследовательский ядерный университет «МИФИ»

Институт интеллектуальных кибернетических систем

Кафедра №12 «Компьютерные системы и технологии»

|  |  |  |
| --- | --- | --- |
| Изображение выглядит как графическая вставка, Графика, графический дизайн, мультфильм  Автоматически созданное описание | ИИКС НИЯУ МИФИ — Институт интеллектуальных кибернетических систем | Изображение выглядит как снимок экрана, Графика, дизайн  Автоматически созданное описание |

**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.

1. **Описание использованных типов данных**

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

1. **Описание использованного алгоритма**

Изображение выглядит как текст, диаграмма, линия, снимок экрана

Автоматически созданное описание

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

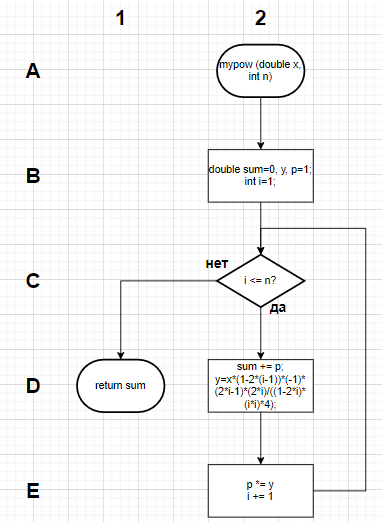


Рис. 2: Блок-схема алгоритма работы функции mypow() программы prog1

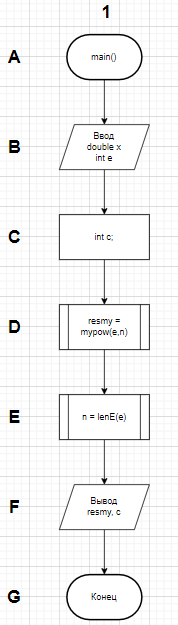


Рис. 3: Блок-схема алгоритма работы функции main() программы prog2

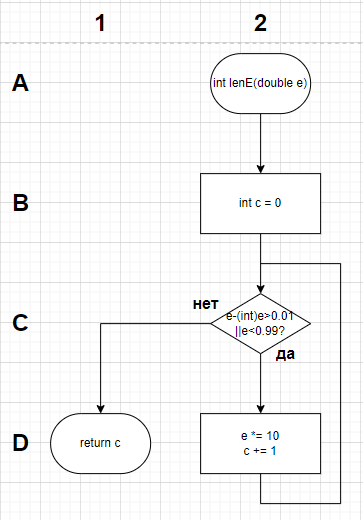
s

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

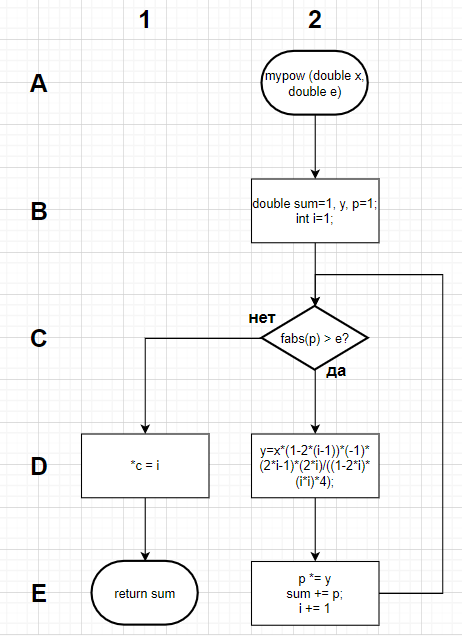


Рис. 5: Блок-схема алгоритма работы функции mypow() программы prog2

1. **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 || x >= 1) {

printf("%s\n","Введите x");

while (scanf("%lf",&x) == 0) {

printf("%s\n","Введите x");

scanf("%\*[^\n]");

}

}

while (n < 1) {

printf("%s\n","Введите n");

while (scanf("%d",&n) == 0) {

printf("%s\n","Введите n");

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;

printf("%lf %d %lf\n", e, (int)e, e-(int)e);

while ((e - (int)e > 0.001) || e < 0.99) {

e\*=10;

c++;

}

return c;

}

int main()

{

double x=-2, e;

int c;

while (x <= -1 || x >= 1) {

printf("%s\n","Введите x");

while (scanf("%lf",&x) == 0) {

printf("%s\n","Введите x");

scanf("%\*[^\n]");

}

}

printf("%s\n","Введите e");

while (scanf("%lf",&e) == 0) {

printf("%s\n","Введите e");

scanf("%\*[^\n]");

}

double resmy = mypow(x,e,&c);

double resmath = pow(x+1,0.5);

int n = lenE(e);

printf("%d\n",n);

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

printf("%s%d\n", "Количество членов ряда: ", c);

printf("%s%.\*lf\n", "Результат библиотеки math: ", n, resmath);

return 0;

}

1. **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 |

1. **Скриншоты**

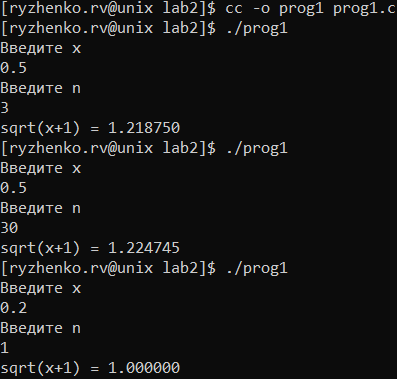
****

Рис. 6: Сборка и запуск программы prog1

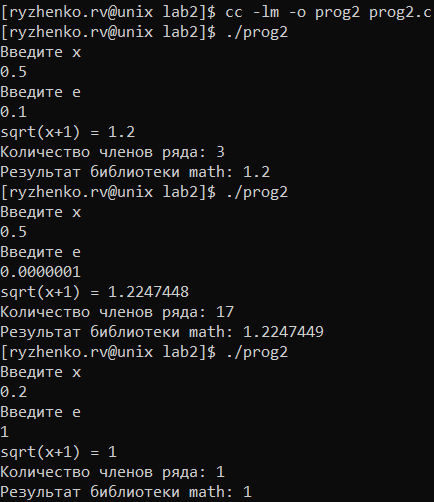


Рис. 7: Запуск программы prog2

1. **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.