ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»  
(РУТ (МИИТ))

Институт транспортной техники и систем управления

Кафедра «Управление и защита информации»

ОТЧЁТ  
О ЛАБОРАТОРНОЙ РАБОТЕ №3.2

По дисциплине «Языки программирования»

ВАРИАНТ 3

Выполнил: ст. гр. ТКИ-142

Величко Иван Сергеевич

Проверил: к.т.н., доц. Васильева М. А.

(Проверил: к.т.н, доц. Балакина Е. П.)

Москва 2024

1. Формулировка задания

Составьте две программы:

a) вычислить сумму первых n членов последовательности (k = 1, 2, 3 ..., n).

b) вычислить сумму всех членов последовательности, по модулю не меньших заданного числа e.

Помните о проверке пользовательского ввода. Все результаты вывести на экран. Отчёт дополнить блок-схемой. При вычислении факториалов необходимо использовать рекуррентное выражение.

Таблица – Исходные данные

|  |  |
| --- | --- |
| Вариант | Ряд |
| 3 |  |

1. Блок-схема алгоритма

Ниже представлены:

Блок-схема основного алгоритма (Рисунок 1).

Блок-схема функции input() (Рисунок 2).

Блок-схема функции input\_positivet() (Рисунок 3).

Блок-схема функции reccurent (Рисунок 4).

Блок-схема функции sum() (Рисунок 5).

Блок-схема функции sum\_e() (Рисунок 6).

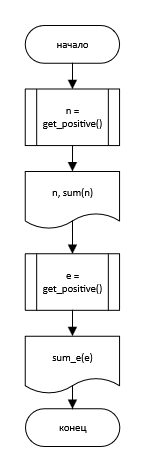


Рисунок – Блок-схема основного алгоритма



Рисунок 2 – Блок-схема функции input()

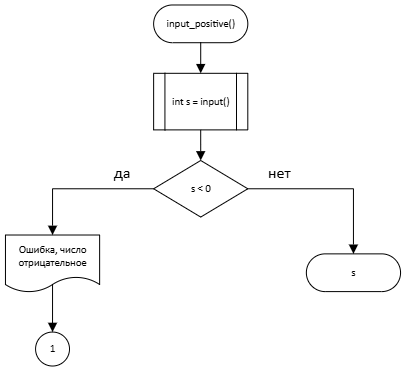


Рисунок – Блок-схема функции input\_positive()



Рисунок – Блок-схема функции reccurent()

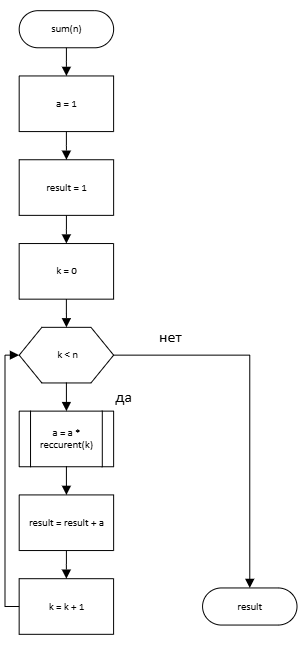


Рисунок – Блок-схема функции sum()

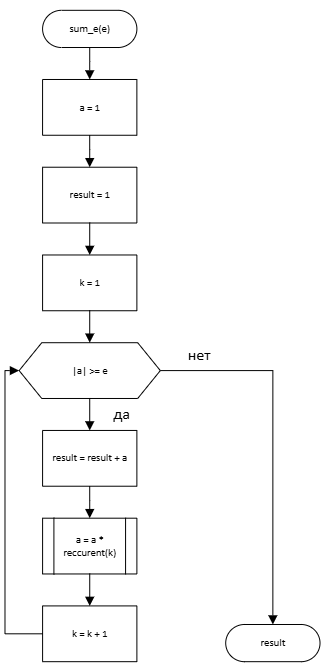


Рисунок – Блок-схема функции sum\_e()

1. Текст программы на языке C

#include <stdio.h>

#include <errno.h>

#include <math.h>

#include <float.h>

#include <stdlib.h>

/\*\*

\* @brief принимает и проверяет числовое значение из stdin

\* @return числовое значение полученное из stdin

\*/

int get\_positive(void);

/\*\*

\* @brief считает сумму первых n элементов последовательности

\* @param n значение числа членов последовательности сумму которых необходимо найти(n)

\* @return сумму первых n элементов последовательности

\*/

double sum(const int n);

/\*\*

\* @brief - вычисляет сумму всех членов последовательности, по модулю не меньших заданного числа e

\* @param e - заданное из stdin число

\* @return сумму всех членов последовательности, по модулю не меньших заданного числа e

\*/

double sum\_e(const int e);

/\*\*

\* @brief возвращает следующий член последовательности по рекуреннтной формуле

\* @param k актуальное порядковое значение элемента последовательности

\* @return следующий член последовательности по рекуреннтной формуле

\*/

double reccurent(const int k);

/\*\*

\* @brief принимает из stdin, а затем возвращает значение переменной value

\* @return числовое значение полученное из stdin

\*/

int input(void);

/\*\*

\* @brief принимает из stdin, а затем возвращает значение переменной value, проверяет является ли число положительным

\* @return положительное числовое значение

\*/

int input\_positive(void);

int main(void) {

puts("Введите чило n:");

int n = input\_positive();

printf("Сумма первых %d элементов последовательности: %f\n", n, sum(n));

puts("Введите чило e:");

int e = input\_positive();

printf("Сумма всех элементов по модулю больше e: %lf", sum\_e(e));

return 0;

}

double sum(const int n) {

double a = 1.0;

double result = 1.0;

for (int k = 0; k < n; k++) {

a \*= reccurent(k);

result += a;

}

return result;

}

double sum\_e(const int e) {

double a = 1.0;

double result = 1.0;

int k = 1;

while (fabs(a) >= e + FLT\_EPSILON) {

result += a;

a \*= reccurent(k);

k++;

}

return result;

}

double reccurent(const int k) {

double value = -1.0 / (k + 1.0);

return value;

}

int input(void) {

double value = 0.0;

int s = scanf\_s("%lf", &value);

if (s != 1) {

errno = EIO;

perror("Ошибка, не числовое значение\n");

exit(1);

}

return value;

}

int input\_positive(void) {

int s = input();

if (s < 0) {

errno = EIO;

perror("Ошибка, число отрицательное\n");

exit(EXIT\_FAILURE);

}

return (size\_t)s;

}

1. Результаты выполнения программы

Ниже представлены:

* Результат выполнения программы (Рисунок 7).
* Результаты тестирования программы (Рисунок 8, Рисунок 9).

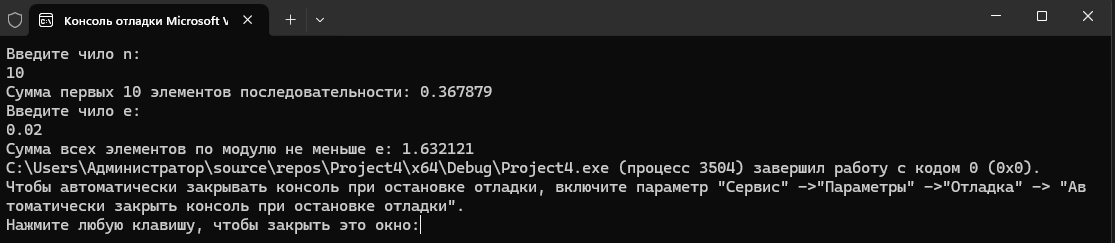


Рисунок – Результат выполнения программы для первых 10 членов последовательности.

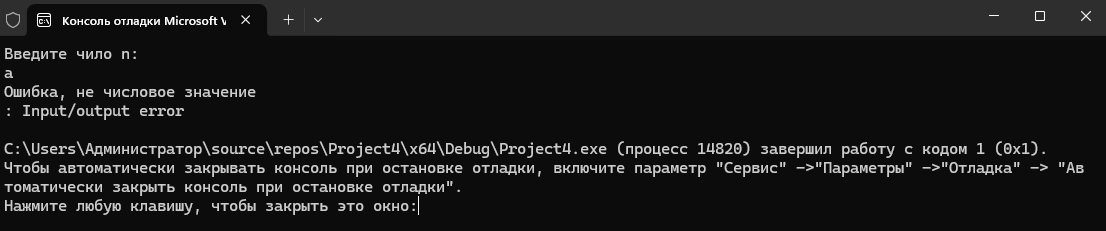


Рисунок – Проверка функции input ()

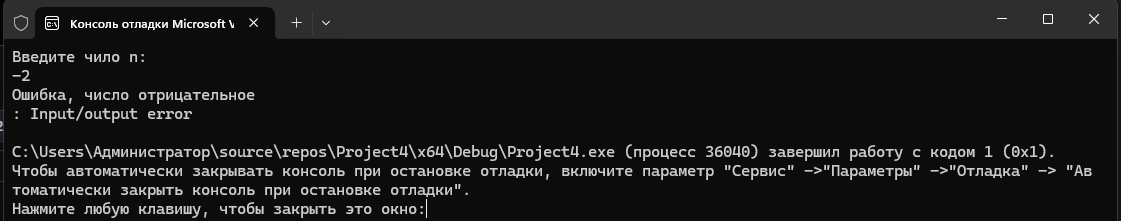


Рисунок 9 – Проверка функции input\_positive()

1. Выполнение тестовых примеров

Результат вычисления суммы первых 10 членов последовательности и суммы всех членов последовательности по модулю не меньше 0.02 в программе MATHLAB приведен ниже (Рисунок 7).

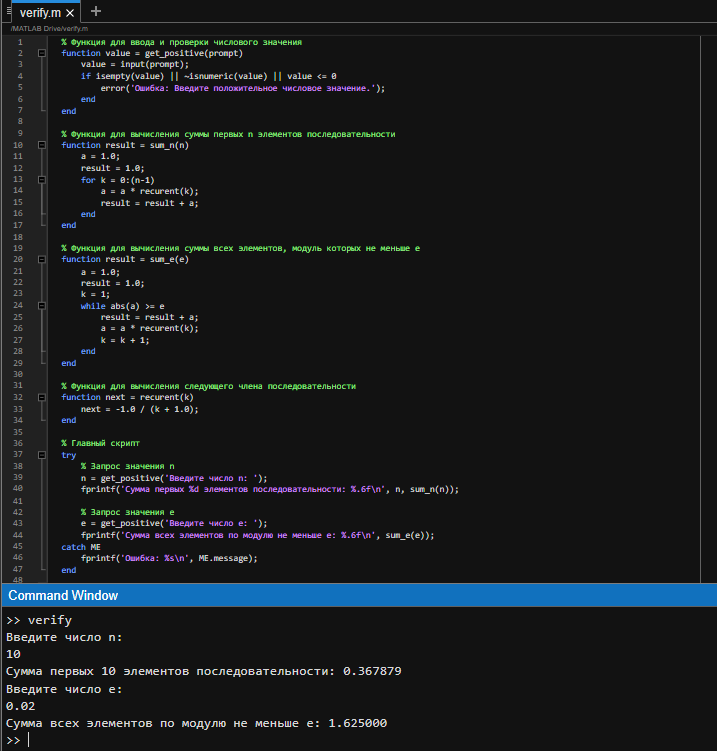


Рисунок – Результат выполнения программы в Mathlab.

1. Отметка о выполнении задания в веб-хостинге системы контроля версий

Отметка о выполнение задания представлена ниже (Рисунок 8)

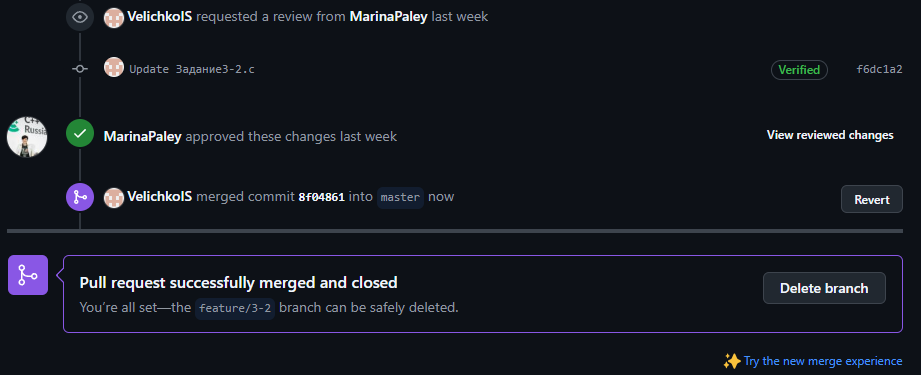


Рисунок – Подтверждение выполнения задания