Министерство науки и высшего образования

Российской Федерации

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ

БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ОБРАЗОВАНИЯ

**«ВЯТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

Факультет автоматики и вычислительной техники

Кафедра радиоэлектронных средств

Отчет по дисциплине

«Цифровые устройства и микропроцессоры»

Лабораторная работа №4

Модульное программирование

Вариант 15

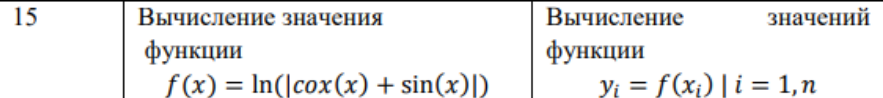
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Выполнил: | студент группы ИНБб-3301 |  | \_\_\_\_\_\_\_\_\_\_\_ |  | Е.Р. Романова |
|  |  |  |  |  |  |
| Проверил: |  |  | \_\_\_\_\_\_\_\_\_\_\_ |  | М.А. Земцов |

Киров 2025

**Цель работы:** знакомство с технологией применения языка ассемблера при разработке программного обеспечения на языках высокого уровня.

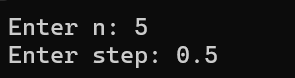
**Ход работы**

**Исходные данные:**

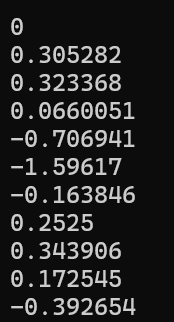


**Верификация:**

Ввод данных:



Результат вычисления значений:



**Пример вычисления значения:**

Помещаем в ebx номер текущей итерации



Вызов функции с текущим элементом



Значение функции



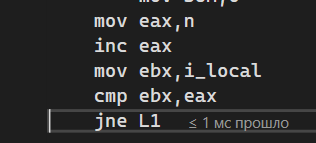
Возврат к ассемблеру. Значение в стеке соответствует рассчитанному значению.



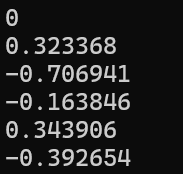
Запись значения в массив



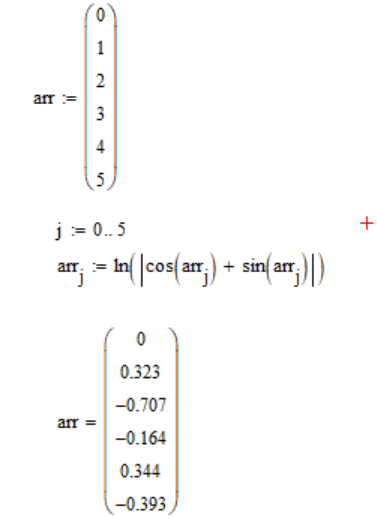
Проверка условия завершения цикла



Вывод всех рассчитанных значений



Результаты, полученные с помощью программы Mathcad



Значения, полученные вручную, совпадают со значениями, полученными с помощью программы.

**Код программы:**

**main.cpp**

#include <iostream>

#include <vector>

using namespace std;

extern "C" float\* SumR(int k, float\* x);

extern "C" float fun\_el(int i, float z) {

float f;

f = log(fabs(cos(z)+sin(z)));

return f;

}

void graph(int n, float step, float\* R) {

int width = n / step;

int height = 80;

for (int y = height - 20; y >= -height; y--) {

for (int x = 0; x < width; x++) {

int outY = static\_cast<int>(R[x] \* height);

if (y == outY) cout << '@';

else if (y == 0) cout << '-';

else if (x == 0) cout << '|';

else cout << ' ';

}

cout << '\n';

}

}

int main(int argc, char\*\* argv) {

int n;

float step;

cout << "Enter n: ";

cin >> n;

cout << "Enter step: ";

cin >> step;

float\* x = new float[n / step + 1];

for (int i = 0; i <= n / step; i++) {

x[i] = i \* step;

}

float\* R;

R = SumR(n / step + 1, x);

cout << endl;

graph(n, step, R);

return 0;

}

**assembly.asm**

.586

.MODEL flat, C

.DATA

SUM DD 0.0

i\_local DD 0

n dd 0

.CODE

extern fun\_el:near

public SumR

SumR proc C

push ebp

mov ebp,esp

mov i\_local,1

mov ecx, dword ptr[ebp+8]

mov n,ecx

mov esi,[ebp+12]

L1:

mov ebx,i\_local

;sub ebx,1

push dword ptr[esi+ebx\*4]

call fun\_el

fld SUM

fadd

inc i\_local

fstp SUM

mov ecx,SUM

mov [esi+ebx\*4],ecx

mov SUM,0

mov eax,n

inc eax

mov ebx,i\_local

cmp ebx,eax

jne L1

mov eax,esi

mov esp,ebp

pop ebp

ret

SumR endp

End