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 №3

|  |  |  |
| --- | --- | --- |
| Выполнил: студент гр. ИКТб-3301-04-00 | \_\_\_\_\_\_\_\_\_\_\_\_ | К.Д. Важенин |
| Проверил: доцент кафедры РЭС | \_\_\_\_\_\_\_\_\_\_\_\_ | М.А. Земцов |

#### Киров 2023

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

**Ход работы:**

**1**.Исходные данные представлены в таблице 1.

Таблица 1 – Исходные данные

Изображение выглядит как текст

Автоматически созданное описание

**2**. Первым действием введем нижний и верхний пределы для определенного интеграла:

int predel\_1, predel\_2;

cout << "Введите нижний предел" << endl;

cin >> predel\_1;

cout << "Введите верхний предел" << endl;

cin >> predel\_2;

Изображение выглядит как текст, снимок экрана, монитор, внутренний

Автоматически созданное описание

После этого выполняется вызов из файла ассемблера функции языка Cи для вычисления значения нижнего предела:

public integral\_calc

integral\_calc proc C

push ebp

mov ebp,esp

push dword ptr [ebp + 8]

call fun\_el

Результат работы функции языка С:

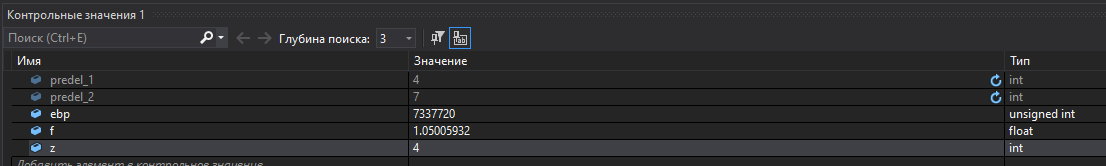
extern "C" float fun\_el(int z)

{float f=0;

f = tanf(z);

f = cbrt(f);

return f;}



Далее значение результата функции языка Си из регистра ST0 записывается в переменную operand1:

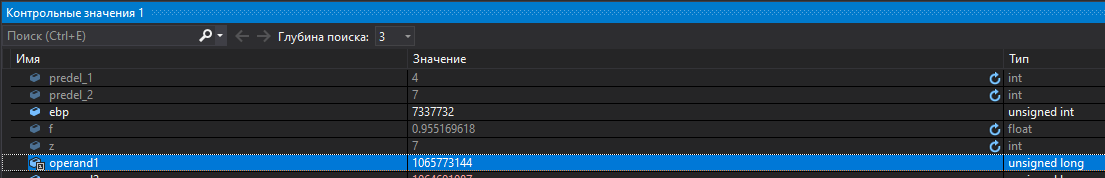
fld operand1

fadd

fstp operand1

push dword ptr [ebp + 12]

call fun\_el



Далее вызов и результат из файла ассемблера функции языка Cи для вычисления значения верхнего предела:

extern "C" float fun\_el(int z)

{

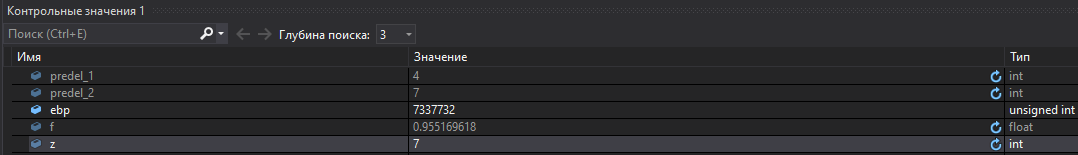
float f=0;

f = tanf(z);

f = cbrt(f);

return f;

}



Потом значение результата функции языка Си из регистра ST0 записывается в переменную operand2. Также записываем разность операндов (operand2 - operand1=результат). Разность значений функции верхнего и нижнего пределов интеграла:

fld operand2

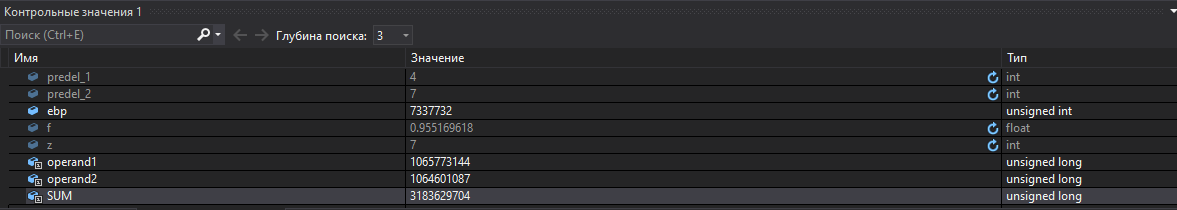
fadd

fstp operand2

fld operand2

fsub operand1

fstp SUM



Последние действии результат вычисления в переменной SUM (регистр ST0) и вывод результата в основном модуле:

fld SUM

mov esp,ebp

pop ebp

ret

integral\_calc endp

double Result = integral\_calc(predel\_1, predel\_2);

cout << "Результат:" << Result << endl;

return 0;

Изображение выглядит как текст, монитор, снимок экрана, внутренний

Автоматически созданное описание

***Вывод:*** в ходе лабораторной работы была ознакомлены с технологией применения языка ассемблера при разработке программного обеспечения на языках высокого уровня.

**3. Код программы:**

#include <iostream>

#include "math.h"

using namespace std;

extern "C" float integral\_calc(int predel\_1, int predel\_2);

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

{

setlocale(LC\_ALL, "Russian");

int predel\_1, predel\_2;

cout << "Введите нижний предел" << endl;

cin >> predel\_1;

cout << "Введите верхний предел" << endl;

cin >> predel\_2;

double Result = integral\_calc(predel\_1, predel\_2);

cout << "Результат:" << Result << endl;

return 0;

}

extern "C" float fun\_el(int z)

{

float f=0;

f = tanf(z);

f = cbrt(f);

return f;

}

Файл ассемблера Source.asm:

.586

.MODEL flat,C

.DATA

operand1 DD 0.0

operand2 DD 0.0

SUM DD 0.0

.CODE

extern fun\_el:near ;

public integral\_calc

integral\_calc proc C

push ebp

mov ebp,esp

push dword ptr [ebp + 8]

call fun\_el

fld operand1

fadd

fstp operand1

push dword ptr [ebp + 12]

call fun\_el

fld operand2

fadd

fstp operand2

fld operand2

fsub operand1

fstp SUM

fld SUM

mov esp,ebp

pop ebp

ret

integral\_calc endp

End