МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

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»

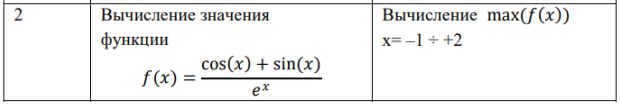
Вариант 2

|  |  |
| --- | --- |
| Выполнил: студент группы ИНБс–3301 | А.И. Харин |
|  |  |
| Проверил: доцент кафедры РЭС | М.А. Земцов |

Киров 2024

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

**Текст задания:**

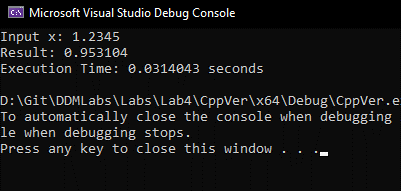


**Ход работы:**

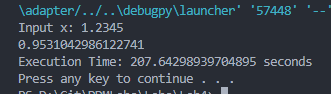
Скорость работы выполнения программы 100000 раз:



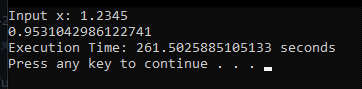
На C++:



На Python интерпретируемый:



На Python компилируемый:



Компилированный код на Python выполнялся дольше интерпретируемого.

**Код программы:**

**Lab4.cpp**

#include <iostream>

#include <math.h>

#include <chrono>

using namespace std;

int main(1) {

int i;

float x, temp, xd;

float f = -99999999;

cout << "Input x: ";

cin >> x;

auto start = chrono::high\_resolution\_clock::now();

for (int j = 0; j < 100000; j++) {

for (i = -1; i < 3; i++) {

xd = x + i;

temp = (cos(xd) + sin(xd)) / exp(xd);

f = max(temp, f);

}

}

cout << "Result: " << f << endl;

auto end = chrono::high\_resolution\_clock::now();

chrono::duration<double> duration = end - start;

cout << "Execution Time: " << duration.count() << " seconds" << endl;

return 0;

}

**AsmFunc.asm**

.586

.XMM

.MODEL flat, C

.DATA

one dd 1 ; Одно для вычитания из x

res dd 0 ; Конечный результат

i dd ? ; Счетчик итераций

temp dd ? ; Временная переменная

xminone dd ? ; Результат вычитания из x

.CODE

extern cfunc:near ; Объявление внешней функции на C++

public getf ; Объявление публичной процедуры

getf proc C x:dword ; Начало процедуры с аргументом x

finit ; Инициализация сопроцессора

fld x ; Загрузка x в стек FPU

fild one ; Загрузка единицы в стек FPU

fsub ; Вычитание из x единицы

fstp xminone ; Сохранение результата вычитания в xminone

yappy:

fld xminone ; Загрузка xminone в стек FPU

fild i ; Загрузка i в стек FPU

fadd ; Сложение

fstp temp ; Сохранение результата в temp

push temp ; Помещение temp в стек вызовов

call cfunc ; Вызов функции на C

fst temp ; Сохранение результата функции

fld res ; Загрузка значения res в вершину стека FPU

fcompp ; Сравнение двух значений в стеке и установка флагов процессора

fstsw ax ; Сохранение состояния флагов в регистре AX

sahf ; Копирование флагов из регистра AH в регистр FLAGS в процессоре

jb move\_to\_tmp ; Если результат больше предыдущего максимального

go\_back:

inc i ; Инкремент счетчика

cmp i, 4 ; Сравнение с 4

jb yappy ; Переход к yappy, если меньше

jmp avoid ; Переход к avoid

move\_to\_tmp:

mov eax, temp ; Перемещение значения temp в eax

mov res, eax ; Перемещение eax в res

jmp go\_back ; Переход к go\_back

avoid:

fld res ; Загрузка res в стек FPU

ret ; Возврат из процедуры

getf endp ; Конец процедуры

End ; Конец файла

**CFunc.cpp**

#include <iostream>

#include <math.h>

using namespace std;

extern "C" float cfunc(float x) {

float f;

cout << "x: " << x << endl; // Вывод значения аргумента

f = (cos(x) + sin(x)) / exp(x); // Вычисление значения функции

cout << "f: " << f << endl; // Вывод значения функции

return f; // Возвращение результата

}

[GitHub репозиторий](https://github.com/khivus/DDMLabs)

**Вывод:** былоприобретено понимание методов интеграции ассемблера в процесс разработки программного обеспечения на языках высокого уровня.