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 №3

«ИСПОЛЬЗОВАНИЕ МАТЕМАТИЧЕСКОГО СОПРОЦЕССОРА»

Вариант №13

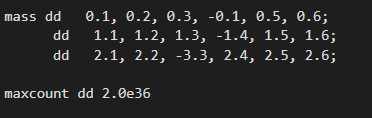
|  |  |  |
| --- | --- | --- |
| Выполнил: студент гр. ИНБб-3301-01-00 | \_\_\_\_\_\_\_\_\_\_\_\_ | А.Н. Пономарев |
| Проверил: доцент кафедры РЭС | \_\_\_\_\_\_\_\_\_\_\_\_ | М.А. Земцов |

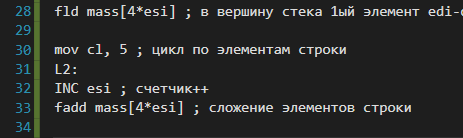
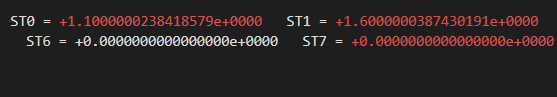
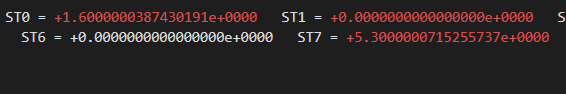
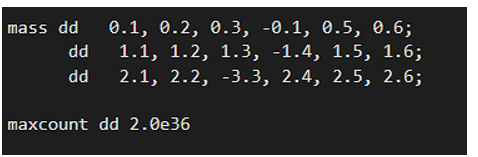
#### Киров 2023

**Цель работы**: изучение принципов выполнения арифметических команд с помощью математического сопроцессора FPU микропроцессоров с архитектурой x86.  
**Вариант задания:** для матрицы N x M найти номер строки, сумма элементов которой является минимальной. Элементы массива 16- разрядные числа со знаком.

**Ход работы:**

1. Объявляем массив, и переменную maxcount. Переменная maxcount, необходима, чтоб загрузить ее на вершину стека. Сумма строк прибавляются к переменной maxcount и в последующем сравниваются.



1. Записываем переменную maxcount в регистр ST0.  
   
2. Складываем переменную maxcount и первое значение массива в первой строке.  
   
3. Складываем получившуюся переменную в прошлой итерации с вторым элементом массива  
   
4. Результат регистра ST0 – сумма первой строки  
   
5. Значение в вершине стека меньше максимального значения, следовательно переходим на метку two. Удаляем st1(maxcount).  
   
6. Переходим на следующую строчку массива.
7. В ST1 помещается прошлая сумма цикла.  
   
8. Проходим циклом по второй строке массива получаем новое значение суммы.
9. Удаляем ST0, прошлое значение записываем в ST7.  
   
10. Суммирования третей строки  
    
11. Переходим на метку more, т.к значение суммы больше найденного до этого. Обнуляем ST0, ST1 в ST0, а после в ST1
12. Результат первой строки:  
    
13. Результат третьей строки:  
    
14. Результат второй строки:  
    
15. В регистре EBP хранится номер строки, сумма элементов которой минимальная.  
    
16. Верефицируем результат, действительно в первой строчке – сумма элементов минимальная.  
    

**Код программы**

.386

.model flat,stdcall

.stack 100h

.data

mass dd 0.1, 0.2, 0.3, -0.1, 0.5, 0.6;

dd 1.1, 1.2,1.3,-1.4,1.5,1.6;

dd 2.1,2.2,-3.3,2.4,2.5,2.6;

maxcount dd 2.0e36

.code

ExitProcess PROTO STDCALL :DWORD

Start:

mov esi, 0 ; счетчик внутри массива

mov ebp, 1 ; номер строки с наименьшим элементом (по умолчанию 1)

finit ; инициализация сопроцессора

mov edi, 1 ; счетчик строки

fld maxcount

L1: ; начало алгоритма

fld mass[4\*esi] ; в вершину стека 1ый элемент edi-ой строки

mov ch, 5 ; цикл по элементам строки

L2:

INC esi ; счетчик++

fadd mass[4\*esi] ; сложение элементов строки

DEC ch

JNZ L2 ; конец цикла по столбцам

fcom st(1)

fstsw ax

sahf

ja one

jb two

je tri

one:

fstp st(0)

jmp konec

two:

fstp st(1)

mov ebp, edi

jmp konec

tri:

fstp st(0)

konec:

INC esi

inc edi

cmp edi, 4

JNZ L1 ; конец цикла по строкам

exit:

Invoke ExitProcess, 1

End Start

**Программа для защиты на СИ**#include <stdio.h>

float mass[] = {

0.1, 0.2, 0.3, -0.1, 0.5, 0.6,

1.1, 1.2, 1.3, -1.4, 1.5, 1.6,

2.1, 2.2, -3.3, 2.4, 2.5, 2.6

};

float maxcount = 2.0e36;

int main() {

int esi = 0;

int ebp = 1;

int edi = 1;

int minSumRow = 1;

float minSum = maxcount;

while (edi <= 3) {

float sum = mass[esi];

int ch = 5;

esi++;

while (ch > 0) {

sum += mass[esi];

ch--;

esi++;

}

if (sum < minSum) {

minSum = sum;

minSumRow = edi;

}

esi++;

edi++;

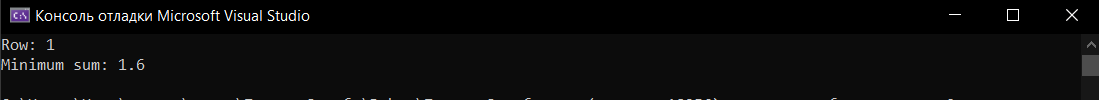
}

printf("Row: %d\n", minSumRow);

printf("Minimum sum: %.1f\n", minSum);

return 0;

}

Верификация:  
  
**Результат идентичен с кодом на ассемблере.**

**Проверка на быстродействие**:

Код на СИ: **14,6 ms.**  
  
  


Код на ассемблере: **9,6 ms**


**Вывод:** код на ассемблере выполняется быстрее на 5 мс.