МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

**(ФГБОУ ВО «КубГУ»)**

**Кафедра информационных технологий**

**Отчет о выполнении лабораторной работы №1**

**по дисциплине «Системы реального времени»**

Работу выполнил студент 45/2 группы А.А. Козин

Руководитель

доц. каф. ИТ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_А.Н. Полетайкин

Краснодар

2023

**ЛАБОРАТОРНАЯ РАБОТА 3**

**ОРГАНИЗАЦИЯ ОБРАБОТКИ ЧИСЛОВЫХ ОДНОМЕРНЫХ МАССИВОВ**

Цель: изучение команд организации циклов и способов косвенной адресации данных памяти в микропроцессоре i486; приобретение практических навыков составления программ обработки одномерных массивов, освоение методов анализа трудоемкости и ресурсной сложности алгоритмов обработки одномерных числовых массивов.

Задания:

1. На основании индивидуального задания (табл. 3.4) составить программу для обработки элементов одномерного массива. Длина элементов исходного массива равна DW. Значения элементов исходного массива задать в сегменте данных (см. пример выше). Длину элементов результирующего массива, если он необходим, выбрать самостоятельно.

2. Получить загрузочный модуль и протестировать выполнение программы.

3. Выполнить расчет времени выполнения программы.

**ХОД РАБОТЫ**

**Задание 1.**

Индивидуальное задание: найти среднее арифметическое элементов массива А[32] и составить новый массив В из элементов, превышающих среднее. Полученный массив В упорядочить по возрастанию, в неотрицательных элементах поменять местами байты.

**Разобью задачу на шаги.**

1. Задать исходный массив A[32] в сегменте данных.
2. Найти среднее арифметическое элементов массива A[32].
3. Создать новый массив B, содержащий элементы, превышающие среднее.
4. Отсортировать массив B по возрастанию.
5. В неотрицательных элементах массива B поменять местами байты.

**Код:**

%include "io.inc"

section .bss

i: resw 1

j: resw 1

avg: resw 1

section .data

A dw 3,4,8,7,6,5,1,2,3,4,8,7,6,5,9,9,9,9,0,0,0,0,6,5,4,3,2,1,1,2,4,4

B dw 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0

len dw 32

section .text

global CMAIN

CMAIN:

mov ebp, esp ; for correct debugging

xor ecx, ecx

xor eax, eax

sum\_loop: ; Цикл для вычисления суммы массива A

xor ebx, ebx

mov bx, [A + 2 \* ecx]

add eax, ebx

; Вычисление среднего значения

inc ecx

cmp ecx, [len]

jl sum\_loop

xor edx, edx

mov ebx, [len]

idiv ebx

mov word[avg], ax

; Сброс значений регистров

xor ecx, ecx

xor ebx, ebx

new\_arr\_loop: ; Цикл для создания нового массива B

xor eax, eax

mov ax, [A + 2 \* ecx]

cmp ax, word[avg]

jle skip\_add

mov [B + 2 \* ebx], ax

inc ebx

skip\_add:

inc ecx

cmp ecx, [len]

jl new\_arr\_loop

; Сортировка массива B по возрастанию

movzx ecx, word[len]

mov ebx, B

i\_loop:

mov word[j], 0

j\_loop:

mov ebx, B

movzx eax, word[j]

add ebx, eax

mov eax, ebx

add eax, 2

mov dx, word[ebx]

mov cx, word[eax]

cmp cx, dx

jl swap

jmp skip

swap:

mov word[ebx], cx

mov word[eax], dx

skip:

add word[j], 2

mov ax, word[len]

add ax, word[len]

sub ax, word[i]

sub ax, 1

cmp [j], ax

jl j\_loop

inc word[i]

mov ax, word[len]

cmp word[i], ax

jl i\_loop

; Изменение байтов неотрицательных элементов в массиве B

xor ecx, ecx

swap\_loop:

xor eax, eax

mov ax, [B + 2 \* ecx]

cmp ax, 0

jle skip\_swap

rol ax, 8

mov [B + 2 \* ecx], ax

skip\_swap:

inc ecx

cmp ecx, [len]

jl swap\_loop

xor ecx, ecx

xor eax, eax

mov edx, [len]

print\_loop:

xor ebx, ebx

mov bx, [B + 2 \* ecx]

PRINT\_DEC 2, bx

PRINT\_CHAR 0x20

inc ecx

cmp ecx, edx

jl print\_loop

mov eax, 1

int 0x80

**Задание 2.**

Контрольные расчеты.

Создам массив A[32].

A =

Найду среднее арифметическое элементов массива A:

Среднее

где:

* N – количество элементов в массиве.
* A[i] – элемент массива A с индексом i.

Подставив значения из массива A, получим:

Среднее

Опустим дробную часть, округлим значение до **4**.

Создам массив В из элементов массива А, которые больше среднего арифметического:

В =

Теперь упорядочим массив В по возрастанию:

В =

Теперь поменяем местами байты в неотрицательных элементах массива В, в 16-битных числах это будет выглядеть так:

Таким образом, новый массив В будет выглядеть следующим образом:

В =

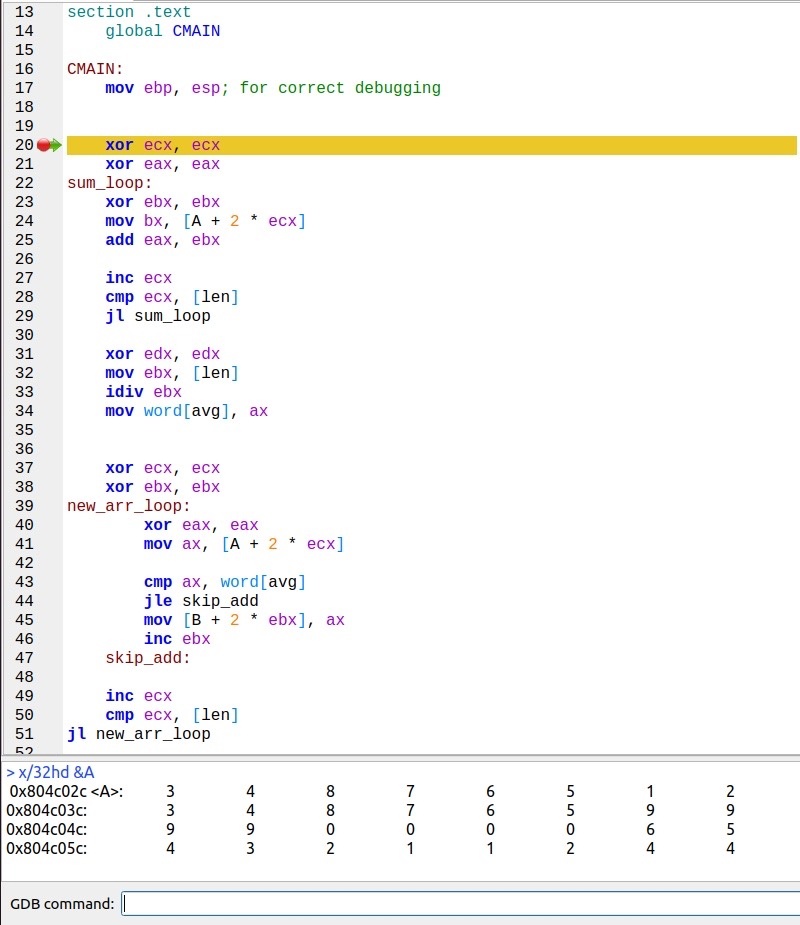


Рисунок 1 – наличие массив A в регистре.

A screenshot of a computer

Description automatically generated

Рисунок 2 – среднее арифметическое массива A.

A screenshot of a computer program

Description automatically generated

Рисунок 3 – массив B с элементами больше среднего арифметического.

A screenshot of a computer

Description automatically generated

Рисунок 4 – отсортированный массив B по возрастанию.

A screenshot of a computer program

Description automatically generated

Рисунок 5 – Итоговый массив B.

**Задание 3.**

Учитывая, что все циклы в худшем случае будут выполняться по 32 раза, а в лучшем 0, имеем 16460–21710 = 3456600–4559100 наносекунд.

**РЕЗУЛЬТАТЫ**

В пункте 1 была проделана работа по созданию кода программы для вычисления арифметического выражения при помощи команд арифметических операций.

В задании 2 была скомпилирована, выполнена и продемонстрирована работа созданной и описанной в п.1.

В задании 3 был проведён расчёт времени выполнения созданного кода.

**ВЫВОДЫ**

Были изучены принципы обработки одномерных массивов и написана программа для выполнения заданий, а также проведён анализ времени работы созданной программы.