МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

**(ФГБОУ ВО «КубГУ»)**

**Факультет компьютерных технологий и прикладной математики**

**Кафедра информационных технологий**

**ОТЧЕТ**

о выполнении лабораторной работы № 3

по дисциплине «Системы реального времени»

Выполнил: ст. гр. 45/2

Посевин Р. Р.

Проверил: доц. каф. ИТ

Полетайкин А.Н.

Краснодар

2021

**ЛАБОРАТОРНАЯ РАБОТА 3**

**Тема:** 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.

**Задание. Вариант 27.**

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.

Задание на обработку: Элементы массива С[17] расположить в порядке убывания. В первых семи элементах нового массива поменять местами байты. Найти минимальный элемент среди нечетных элементов массива.

**Ход работы**

Определим значения массива A в сегменте данных (рисунок 1).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 1 - представление массива в сегменте данных

Ниже представлен листинг программы для обработки элементов одномерного массива.

.686

include \masm32\include\io.asm

.data

A DW 7, -3, 5, 2, 4,

8, -11, 27, 6, 4,

11, 1, 12, 55, 16,

-77, -21

N DW 17

MIN DW 0

M DW 1000 dup(0)

.code

LStart:

; array sort

MOV CX, N

DEC CX

XOR EDI, EDI

outer\_cycle:

PUSH CX

MOV ESI, EDI

inner\_cycle:

ADD ESI, 2

MOV AX, [A][EDI]

MOV BX, [A][ESI]

CMP AX, BX

JGE skip\_replace

MOV [A][EDI], BX

MOV [A][ESI], AX

skip\_replace:

DEC CX

JNZ inner\_cycle

ADD EDI, 2

POP CX

DEC CX

JNZ outer\_cycle

; replace bites in first 7 numbers

MOV CX, 7

XOR EDI, EDI

replace\_cycle:

MOV AX, [A][EDI]

MOV BH, AH

MOV AH, AL

MOV AL, BH

MOV [A][EDI], AX

ADD EDI, 2

DEC CX

JNZ replace\_cycle

; find min uneven number

MOV BX, A

MOV CX, N

DEC CX

MOV EDI, 2

min\_cycle:

MOV AX, [A][EDI]

TEST AX, 1

JZ continue\_min\_cycle

CMP AX, BX

JG continue\_min\_cycle

MOV BX, AX

continue\_min\_cycle:

ADD EDI, 2

DEC CX

JNZ min\_cycle

MOV MIN, BX

end LStart

Протестируем работу программы при помощи дебаггера. На рисунке 2 представлен дамп памяти на момент начала работы программы. Теперь последовательно выполним все действия программы. На рисунках 3 и 4 представлен дамп памяти в ключевые моменты работы программы. На рисунке 5 представлен дамп памяти на момент окончания работы программы.

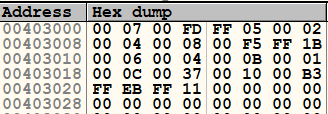


Рисунок 2 – Дамп памяти перед началом работы программы

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 3 – Дамп памяти после сортировки массива

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 4 – Дамп памяти после перестановки байтов в первых 7 элементах массива

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 5 – Дамп памяти на момент окончания работы программы

На рисунке 5 можем увидеть минимальный нечетный элемент массива, записанный после массива и количества элементов в нем.

B3FFh = -77

Исходя из рисунков можно сделать вывод, что программа работает корректно.

Посчитаем время выполнения программы (Рисунок 6).

Изображение выглядит как текст

Автоматически созданное описание

Исполнение программы занимает около 13850 тактов процессора.

Частота процессора:

t = 1 / 3 200 000 000 = 0,0000000003125 с.

Время в наносекундах:

N \* t = 0,0000000003125 \* 13850 = 0,000004328125 c = 4,33 мкс.

Время выполнения программы составляет 4,33 мкс.

**Вывод**

В ходе работы получены базовые навыки обработки одномерных массивов целых чисел, построения алгоритмов с использованием ветвления и множественного перебора элементов массива, вывода массива, а также циклов, условных и безусловных переходов к меткам.