МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ   
РОССИЙСКОЙ ФЕДЕРАЦИИ   
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ОРЛОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ И.С. ТУРГЕНЕВА»

Кафедра информационных систем

Отчет   
по лабораторной работе №4  
на тему: «Обработка массивов. Числа Фибоначчи»  
по дисциплине «Архитектура ЭВМ и систем»

Выполнили: Марочкин М.А. Шифр: 170584  
 Яшин М. О. Шифр: 170133  
 Шорин В.Д. Шифр: 171406  
ИПАИТ  
Направление: 09.03.04 «Программная инженерия»  
Группа: 71-ПГ  
Проверил(а):\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
Отметка о зачете:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
 Дата «\_\_\_\_» \_\_\_\_\_\_ 2017 г.

Орел, 2017г.

**Цель работы.**

Цели лабораторной работы:

1. Изучение принципов функционирования памяти и микропроцессора компьютера при выполнении операций над массивами данных.

2. Приобретение навыков использования команд ассемблера, связанных с обработкой массивов.

3. Получение представления об особенностях обработки данных, команд и режимах доступа к данным при обработке массивов.

**Контрольные вопросы:**

1. Массивы и их представление в памяти компьютера.

Массив – это последовательность элементов, доступ к которым осуществляется при помощи целочисленных индексов, следующих по порядку.

Все элементы массива располагаются в памяти компьютера последовательно. Одну и ту же область памяти можно трактовать одновременно и как одномерный, и как двухмерный массив. Все зависит только от алгоритма обработки этих данных в конкретной программе.

2. Режимы адресации данных, которые могут применяться для доступа к элементам массива.

*Регистровый относительный.* Является обобщением методов адресации, обеспечивающих вычисление эффективного адреса (EA) операнда в памяти в виде суммы базового значения адреса и «смещения» disp, указываемого в команде.

*Индексный.* Применяется для обработки упорядоченных массивов данных; при этом каждый элемент массива определяется собственным номером. Тогда базовый адрес массива задаётся смещением disp, указываемым в команде, а значение индекса (номер элемента массива) определяется содержимым индексного регистра.

*Базово-индексный.* Используется для доступа к элементам массива, адресуемого указателем. Базовый адрес массива задаётся указателем базы (базовым регистром), а номер элемента массива – содержимым индексного регистра.

*Относительный базовый индексный.* Используется для адресации элементов в указываемом массиве записей. Базовый адрес массива задаётся указателем базы, номер записи (т.е., элемента массива) определяется содержимым индексного регистра, а смещение в команде указывает расстояние до записи.

3. Способы описания массивов в сегменте данных.

При необходимости использовать массив в программе его нужно описать одним из следующих способов:

Перечислением элементов массива в поле операндов одной из директив описания данных. При перечислении элементы разделяются запятыми.

mas db 1, 2, 3, 4, 5, ’$’

Используя оператор повторения dup.

mas db 5 dup (?)

4. Особенности обработки двумерных массивов в ассемблерных программах. Вычисление смещения элемента двумерного массива относительно начала сегмента данных.

Двумерный массив нужно моделировать. На описании самих данных это почти никак не отражается — память под массив выделяется с помощью директив резервирования и инициализации памяти.

Непосредственно моделирование обработки массива производится в сегменте кода, где программист, описывая алгоритм обработки ассемблеру, определяет, что некоторую область памяти необходимо трактовать как двухмерный массив.

Смещение элемента определяется по формуле:

база + размер\_элемента \* (кол-во\_элементов\_в\_строке \* i + j), где

i – номер строки от 0 до n-1

j – номер столбца от 0 до m-1

5. Какие режимы адресации данных можно использовать для доступа к элементам двумерного массива?

Базовая индексная адресация со смещением – режим адресации, при котором эффективный адрес формируется максимум из трех компонентов:

* постоянного (необязательный компонент), в качестве которого может выступать прямой адрес массива в виде имени идентификатора, обозначающего начало массива, или непосредственное значение;
* переменного (базового) – указанием имени базового регистра;
* переменного (индексного) – указанием имени индексного регистра.

Этот вид адресации удобно использовать при обработке двумерных массивов.

**Текст программы.**

d\_s segment

mas dw 18 dup (?) ; определяем массив на 18 двухбайтовых ячеек

min dw 0

max dw 0

d\_s ends

assume ds: d\_s, cs: c\_s

c\_s segment

begin:

mov ax, d\_s

mov ds, ax

mov si, 0

mov mas [si], 0

add si, 2

mov mas [si], 1

mov cx, 16

met1:

add si, 2

mov ax, mas [si - 4]

mov bx, mas [si - 2]

add ax, bx

mov mas [si], ax

mov dx, mas[si]

loop met1

mov dh, 1 ; заносим в dh номер строки первого элемента второй строки

mov dl, 0 ; заносим в dl номер столбца первого элемента второй строки

mov al, 12

mul dh

mov bx, ax

mov al, 2

mul dl

add ax, bx ; в ax смещение первого элемента второй строки

mov dx, 500 ; определяем максимальное значение

mov si, ax

mov cx, 6

met2:

mov ax, mas [si]

shr ax, 1

jc met3

jmp met4

met3: shl ax, 1

add ax, 00000001b

cmp ax, dx

jl met5

jmp met4

met5: mov dx, ax

met4: add bl, 1

add si, 2

loop met2

mov min, dx ; заносим минимальное значение в переменную min

mov dh, 0 ; заносим в dh номер строки четвертого элемента первой строки

mov dl, 3 ; заносим в dl номер столбца четвертого элемента первой строки

mov al, 12

mul dh

mov bx, ax

mov al, 2

mul dl

add ax, bx ; в ax смещение четвертого элемента первой строки

mov dx, 0 ; определяем минимальное значение

mov si, ax

mov cx, 3

met6:

mov ax, mas [si]

shr ax, 1

jc met7

shl ax, 1

cmp ax, dx

jg met8

jmp met7

met8: mov dx, ax

met7: add bl, 1

add si, 12

loop met6

mov max, dx ; заносим максимальное значение в переменную max

mov cx, min ; заносим минимальное значение в cx

mov ah, 4ch

int 21h

c\_s ends

end begin