**Работу выполнил Кузичкин Павел**

**Лабораторная работа №4**

Машинно-зависимые языки

Текст программы:

.model small

.stack 100h

.186

.data

array dw 20 dup (0)

res1 db 5 dup (0), '$'

res2 db 5 dup (0), '$'

.code

start:

mov ax, @data

mov ds, ax

mov ax, 0

mov si, 0

mov cx, 10

mov bl, 5

push cx

first:

mov array[si], bx

add si, 2

add bl, 5

loop first

pop cx

push cx

mov si, 0

second:

mov ax, array[si]

mul ax

mov array[si+20], ax

add si, 2

loop second

pop cx

mov si, 0

mov bx, 10

push cx

call ShowFirstLine

mov dl, 0ah

mov ah, 02h

int 21h

pop cx

call ShowSecondLine

mov ax, 4c00h

int 21h

byte\_asc proc

pusha

mov si, 4

label2:

mov dx, 0

div bx

add dl, 30h

mov res1[si], dl

dec si

cmp ax, 0

jnz label2

popa

ret

byte\_asc endp

byte\_asc1 proc

pusha

mov si, 4

label3:

mov dx, 0

div bx

add dl, 30h

mov res2[si], dl

dec si

cmp ax, 0

jnz label3

popa

ret

byte\_asc1 endp

ShowFirstLine proc

label1:

mov ax, array[si]

call byte\_asc

mov ah, 09h

mov dx, offset res1

int 21h

add si, 2

loop label1

ret

ShowFirstLine endp

ShowSecondLine proc

label0:

mov ax, array[si]

call byte\_asc1

mov ah, 09h

mov dx, offset res2

int 21h

add si, 2

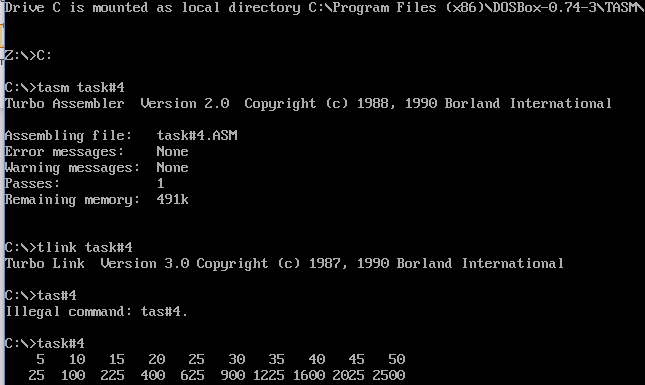
loop label0

ret

ShowSecondLine endp

end start

Результат выполнения программы:



Ответы на контрольные вопросы:

1. Какой командой можно выделить в памяти место под одномерный массив байтов array размерностью 25?

Ответ:mas DB 25 DUP (0);массив на 25 эл-тов

1. Опишите команды умножения на байт и на слово.

Ответ:

Команда умножения MUL

MUL 2-й множитель

1-й множитель будет выбран в зависимости от размера 2-го множителя (это будет AL или AX).

По команде MUL умножаются два целых числа без знака, при этом результат тоже не имеет знака. На приведенных ниже рисунках представлены варианты команд умножения с байтами и со словами.

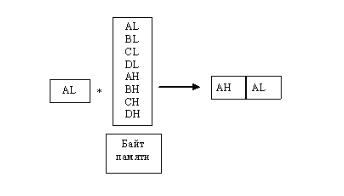


Рис.4.9. Умножение байт.

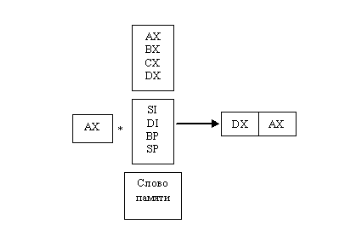


Рис.4.10. Умножение слов.

При умножении 8-битовых операндов результат всегда помещается в регистр AX . При умножении 16-битовых данных результат, который может быть длиною до 32 бит, помещается в пару регистров: в регистре DX содержатся старшие 16-бит, а в регистре AX – младшие 16-бит. Умножение не допускает непосредственного операнда.

Установка флагов командой умножения отличается от других арифметических команд. Единственно имеющие смысл флаги – это флаг переноса и переполнения.

Команда MUL устанавливает оба флага, если старшая половина результата не нулевая. Если умножаются два байта, установка флагов переполнения и переноса показывает, что результат умножения больше 255 и не может содержаться в одном байте. В случае умножения слов флаги устанавливаются, если результат больше 65535.

1. Какое максимальное беззнаковое число можно хранить в элементе массива размером в 1 байт?

Ответ:255

1. Пусть имеется массив: array DW 50 DUP(?). Для доступа к отдельным элементам массива используется адресное выражение array[SI]. Как называется этот способ адресации и как с его помощью будет вычисляться адрес элементов массива?

Ответ:

***индексная адресация со смещением*** — режим адресации, при котором эффективный адрес формируется из двух компонентов:

**постоянного (базового)** — указанием прямого адреса массива в виде имени идентификатора, обозначающего начало массива;

**переменного (индексного)** — указанием имени индексного регистра

1. Каким образом осуществляется перебор элементов некоторого массива A с помощью адресного выражения A[SI], если массив состоит из байтов, слов или двойных слов?

Ответ:При переборе к смещению прибавляется 1/2/4

1. Для некоторого массива A каким будет результат выполнения команды mov DI, A и команды mov DI, offset A?

Ответ:в DI будет помещён элемент с адресом, который хранится в DS/будет записано смещение на адрес массива А.