

Разбор контрольной работы № 1

№ 1	Процессоры Intel x86. Часть 1
1. Перечислить все 8-битные общие регистры. Какие из них выполняют специальные функции?	
2. Задан фрагмент программы. Определить способы адресации операндов в каждой команде. varC equ dword ptr[bp+4]; адрес параметра C push bp ; сохранение bp mov bp,sp; настройка bp на вершину стека les bx,varC; es:[bx]:=адрес C mov al,es:[bx]; al:=символ cmp al,'a' ; символ меньше "a"?	
3. Какие из приведенных ниже команд являются недопустимыми? Указать ошибку в каждой недопустимой команде. Считать все имена переменными, которые определены как слова: а) MOV [BX+4*3], SP; б) MOV WORD_OP1, WORD_OP2; в) MOV AX, WORD_OP [DX]; г) MOV SP, SS: [BX+SI]; д)MOV CS, AX;	
4. По каким адресам памяти сохраняется регистр ESI при выполнении команды PUSH ESI в реальном режиме, если (SS) = 6400H и (SP) = 1FF0H?	
5. Следующий фрагмент программы записать без использования строковых команд: <i>cld</i> <i>rep movsb</i>	

Задание 2.

Задан фрагмент программы. Определить способы адресации операндов в каждой команде.

```
varC equ dword ptr[bp+4]; адрес параметра C
push bp ; сохранение bp
mov bp,sp; настройка bp на вершину стека
les bx,varC; es:[bx]:=адрес C
mov al,es:[bx]; al:=символ
cmp al,'a' ; символ меньше "a"?
```

Команда	Адресация	
	первого операнда	второго операнда
push bp	регистровая	по умолчанию
mov bp,sp	регистровая	регистровая
les bx, dword ptr[bp+4]	регистровая	косвенная (базовая)
mov al,es:[bx]	регистровая	косвенная
cmp al,'a'	регистровая	непосредственная
mov bx,fs:[task_counter]	регистровая	прямая

Задание 3

Какие из приведенных ниже команд являются недопустимыми? Указать ошибку в каждой недопустимой команде. Считать все имена переменными, которые определены как слова:

- а) MOV [BX+4*3], SP;
- б) MOV WORD_OP1, WORD_OP2;
- в) MOV AX, WORD_OP [DX];
- г) MOV SP, SS: [BX+SI];
- д) MOV CS, AX

Команда	Ошибка
MOV [BX+4*3], SP	нет
MOV WORD_OP1, WORD_OP2	оба операнда не могут быть ячейками памяти
MOV AX, WORD_OP [DX]	регистр DX нельзя использовать для адресации
MOV SP, SS: [BX+SI];	нет
MOV CS, AX	регистр CS нельзя загружать командой MOV

Как проверить в программе на ассемблере?

```
data segment
```

```
WORD_OP1 dw 5
```

```
WORD_OP2 dw 10
```

```
WORD_OP dw 20
```

```
data ends
```

```
code segment
```

```
    assume cs: code, ds: data
```

```
    mov ax, data
```

```
    mov ds, ax
```

```
    MOV [BX+4*3], SP;
```

```
    MOV WORD_OP1, WORD_OP2;
```

```
    MOV AX, WORD_OP [DX];
```

```
    MOV SP, SS: [BX+SI];
```

```
    MOV CS, AX
```

```
code ends
```

```
end
```

main1.ASM

```
1  0000                                data segment
2  0000 0005                            WORD_OP1 dw 5
3  0002 000A                            WORD_OP2 dw 10
4  0004 0014                            WORD_OP dw 20
5  0006                                data ends
6
7  0000                                code segment
8                                        assume cs: code, ds: data
9  0000 B8 0000s                          mov ax, data
10 0003 8E D8                             mov ds, ax
11 0005 89 67 0C                          MOV [BX+4*3], SP;
12 0008 8C D6 0000r                       MOV WORD_OP1, WORD_OP2;
**Error** main1.ASM(12)                   Need register in expression
13 000C 8B                                MOV AX, WORD_OP [DX];
**Error** main1.ASM(13)                   Illegal indexing mode
14 000D 36: 8B 20                          MOV SP, SS: [BX+SI];
15                                        MOV CS, AX
**Error** main1.ASM(15)                   Invalid operand(s) to instruction
16 0010                                code ends
17                                        end
```

Задание 4

По каким адресам памяти сохраняется регистр ESI при выполнении команды PUSH ESI в реальном режиме, если (SS) = 6400H и (SP) = 1FF0H?

Размер регистра ESI 4 байта, поэтому регистр SP уменьшается на 4 и в новую вершину стека копируется регистр SI.

$$SP - 4 = 1FECh$$

Логический адрес (Сегмент: Смещение) новой вершины стека
6400h: 1FECh.

Физический адрес в реальном режиме формируется по формуле
Сегмент * 16 + Смещение.

$$\text{Адрес} = 6400h * 10h + 1FECh.$$

$$64000$$

$$\underline{+01fec}$$

$$65fec$$

Таким образом, регистр ESI сохраняется в памяти в 4-х ячейках памяти:
65fec, 65fedh, 65feeh, 65fefh, где первым указан адрес младшего байта.

Задание 5 (устные комментарии)

Следующий фрагмент программы записать без использования строковых команд:

```
cld  
rep movsb
```

(задача на строковые команды)

Пусть X, Y – переменные, имеющие тип byte. Написать фрагмент программы, вычисляющей выражение:

$X * Y / 10 + 25$

Результат сохранить в AX.

(Задача на арифметические команды).

Написать фрагмент программы для обнуления 8-х старших бит, установки 4-х младших бит и инвертирования бит 8-11 регистра ESI без изменения остальных бит.

(Задача на логические команды, можно использовать AND/OR/XOR или VTR/BTS/BTC).

Написать фрагмент программы, которая в 30 последовательных слов памяти, начиная с метки X, заносит квадраты 30 натуральных чисел от 1 до 30.

(Задача на организацию циклов и косвенную адресацию памяти).