

作业解答及总复习

第一次作业：

P30 习题一 1.1, 1.2, 1.7 3.1, 3.2, 3.4

3.1 COUNT 不分配存储空间

STR1	0	0
	1	
	2	
	3	
	4	
	5	
STR2	30H('0')	6
	31H('1')	
	32H('2')	
	33H('3')	
	34H('4')	
	35H('5')	
NUMB	2	12 (0CH)
	2	
	2	
NUMW	10	15(0FH)
	0	
	0A0H	
	0FFH	
POIN	0	19(13H)
	0	

3.2

解答：变量的偏移地址：变量的偏移地址是指定义变量所在段的段首址到该变量所占存储单元的字节距离。

变量的值：变量所指向的存储单元中存储的数据。

STR1 的偏移地址是：0000H， 变量 STR1 的值是 0， 变量 STR1+1 的值是 1， 变量 STR1+2 的值是 2 ， 变量 STR1+3 的值是 3 ， 变量 STR1+4 的值是 4， 变量 STR1+5 的值是 5， STR2 的偏移地址是：0006H， 变量 STR2 的值是 30H， 变量 STR2+1 的值是 31H， 变量 STR2+2 的值是 32H ， 变量 STR2+3 的值是 33H ， 变量 STR2+4 的值是 34H， 变量 STR2+5 的值是

35HNUMB 的偏移地址是: 000CH, 变量 NUMB 的值是 02H , 变量 NUMB+1 的值 02H, 变量 NUMB+2 的值 02H NUMW 的偏移地址是: 000FH, 变量 NUMW 的值是 0010H, 变量 NUMW+2 的值是 0FFA0H, POIN 的偏移地址是: 0013H, 变量 POIN 的值是 0000H

第二次作业

1.教材 P31 1.14

已知 8 位二进制数 x_1 和 x_2 的值, 求 $[x_1]_{\text{补}} + [x_2]_{\text{补}}$, 并指出结果的符号, 判断是否产生了溢出和进位。

$$(1) \quad x_1 = +0110011B \quad x_2 = +1011010B$$

$$(2) \quad x_1 = -0101001B \quad x_2 = -1011101B$$

$$(3) \quad x_1 = +1100101B \quad x_2 = -1011101B$$

解答: (1)

$$[x_1]_{\text{补}} + [x_2]_{\text{补}} = 00110011B + 01011010B = 10001101B$$

$$\begin{array}{r} 00110011 \\ +) 01011010 \\ \hline 10001101 \end{array}$$

产生溢出 $OF=1$, 没有产生进位 $CF=0$ (两正数相加得到一个负数, 产生溢出)

$$(2) \quad [x_1]_{\text{补}} = 11010111B \quad [x_2]_{\text{补}} = 10100011B$$

$$-0101001B \rightarrow \text{相反数 } 00101001B \quad \text{连同符号位一起取反加 } 1 \rightarrow 11010111B$$

$$\begin{array}{r} 11010111B \\ +) 10100011B \\ \hline 01111010B \end{array}$$

$$(3) \quad x_1 = +1100101B \quad x_2 = -1011101B$$

$$[x_1]_{\text{补}} = 01100101B \quad [x_2]_{\text{补}} = 10100011B$$

$$\begin{array}{r} 01100101B \\ +) 10100011B \\ \hline 0001000B \end{array}$$

没有溢出 $OF=0$, 有进位 $CF=1$

2. 已知表首址为 TAB, TAB 定义为:

TAB DW 30H, 41H, 42H, 43H, 44H, 45H, 46H

写出下列各语句执行后, (AL)=?, (CL)=?, (DX)=?

```
MOV ESI, 1  
MOV AX, TAB[ESI*2];  
ADD ESI, 3  
MOV CX, TAB[ESI*2];  
MOV DX, TAB[ESI];
```

若第二条语句修改为: **MOV AX, TAB[ESI];** 则执行该语句之后(AX)=?

若第二条语句修改为: **MOV AX, TAB[ESI];** 则执行该语句之后(AX)=?

解答: 执行各语句后: (AL) =41H (CL) =44H (DX) =42H

TAB	30H
	0
	41H
	0
	42H
	0
	43H
	0

下面是解答过程:

```
MOV ESI, 1 (ESI)=1  
MOV AX, TAB[ESI*2]; (AX) =41H→(AL)=41H  
ADD ESI, 3 ; (ESI) =4  
MOV CX, TAB[ESI*2]; (CX)=44H→(CL)=44H  
MOV DX, TAB[ESI]; (DX)=42H
```

若第二条语句修改为: **MOV AX, TAB[ESI];** 则执行该语句前 (ESI) =1, 执行后(AX)=4100H

3. 已知数据段定义如下：

```

DATA SEGMENT USE16
BUF1 DW 3435H
BUF2 DB 'AB','$'
CON EQU 30 ; 不分配空间
BUF3 DW BUF1
DB 2 DUP (9)
BUF4 DW —60H
DATA ENDS
    
```

(1)请以字节为单位填写数据在存储器中的存放形式，并在图中标明 BUF1、BUF2、BUF3 和 BUF4 所处的偏移地址值。

(2)假设在执行下面的指令之前，已经将 DATA 段首地址赋与 DS 段寄存器执行指令：MOV BX， BUF3+3 后 (BX)=?

解答：(1)如下图所示。

(2) 执行 MOV BX, BUF3+3 后(BX)=0A009H

BUF1	35H	0
	34H	
BUF2	41H(或'A')	2
	42H(或'B')	
	24H(或'\$')	
BUF3	0	5
	0	
	9	
	9	
BUF4	0A0H	9
	0FFH	

第三周作业:

1. 现有下列程序段

```
MOV     SI, OFFSET BUF1
MOV     DI, OFFSET BUF2
MOV     CX, n
LOOPA:  MOV     AL, [SI]
        MOV     [DI], AL
        INC     SI
        INC     DI
        LOOP    LOOPA  (CX)=0FFFFH
```

回答如下 (1) 至 (5) 问题。其中 BUF1、BUF2 均为串长度为 n 的字节存储区首址。

- (1) 该程序段完成了什么工作?
- (2) 若将指令“MOV CX, n”误写成“MOV CX, 0”, 则循环体被执行多少次?
- (3) 若漏掉了指令“MOV CX, n”, 则循环体执行的次数能确定吗? 为什么?
- (4) 若漏掉了指令“INC SI”, 则程序运行果如何?
- (5) 若不小心将标号 LOOPA 上移了一行, 即将标号写在指令“MOV CX, n”之前, 则程序运行情况?

解答: (1) 该程序完成将 BUF1 存储区中的 n 各字节转储 (或者拷贝) 到 BUF2 存储区

(2) 执行 65536 次

(3) 不能。这段程序中没有给 CX 赋初始值, 因此循环次数不确定

(4) 若漏掉“INC SI”指令语句, 则将 BUF1 存储区第一个字节复制到 BUF2 的 n 个字节中

(5) n=1, 执行一次, n>1 程序将形成死循环。

2. 编写一段程序段, 调用基数转换子程序将 EAX 中的有符号二进制数转换后放在以 SI 为首地址的存储区中, 子程序结束时 SI 指向转换后字符串的末尾。

子程序入口参数: EAX, 待转换的有符号二进制数 EBX: 转换的进制 SI: 缓冲区首址

```
YFH  PROC
      OR  EAX, EAX
      JGE L1 ---JGE/JNS 都可以, 还可以是 TEST EAX,EAX  JGE/JNS
L1/AND EAX, EAX/CMP EAX, 0  JGE/JNS, 有多种写法
      MOV [SI], BYTE PTR '-'
      INC SI
      NEG EAX
L1:   CALL RADIX
      RET
YFH  ENDP
```

第四周作业:

1.定义一个宏 EXPA, 其调用形式是 EXPA A1, B1, C1(其中 A1, B1, C1 均为有符号字类型变量), 该宏实现计算 $(A1*5+B1)/8$ 的值, 并保存在 C1 中, 不考虑溢出。

解答: EXPA MACRO A1, B1, C1

MOV AX, A1

SAL AX, 2

ADD AX, A1

ADD AX, B1

SAR AX, 3

MOV C1, AX

ENDM

此外还有其他做法: 比较普遍的是:

EXPA MACRO A1, B1, C1

IMUL AX, A1, 5;; 或: MOV AX, A1 IMUL AX, 5

ADD AX, B1

CWD

MOV BX, 8

IDIV BX (DX, AX)

MOV C1, AX

ENDM

2.设学生结构和数据段定义如下:

STUDENT STRUCT

XUEHAO DW 0

STU_NAME DB 'ZHANG'

YUWEN DB 0; 语文成绩

SHUXUE DB 0; 数学成绩

ENGLISH DB 0; 英语成绩

ZONGFEN DW 0; 总分

STUDENT ENDS

.DATA

S1 STUDENT <50, , 74, 95, 80,>

编写程序段计算变量 S1 中三门课程的总分, 并保存在 ZONGFEN 字段中。

MOVZX AX, S1.YUWEN 比较多的学生未作扩展

MOVZX BX, S1.SHUXUE

ADD AX, BX

MOVZX BX, S1.ENGLISH

ADD AX, BX

MOV S1.ZONGFEN, AX

第五周作业

1. 写出从外部设备 71H 端口读入一个字节数据保存在字节变量 BB 中的程序段。

```
IN AL, 71H
```

```
MOV BB, AL
```

2. 设在实方式下，有一个子程序名为 INT_PROC 的中断处理程序，现需要将该程序设置成 45H 号中断，写出完成该设置功能的程序段。

```
MOV AX, 0
MOV DS, AX
CLI
MOV BX, OFFSET INT_PROC
MOV DS:[45H*4], BX
MOV BX, SEG INT_PROC
MOV DS:[45H*4+2], BX
STI
```

```
MOV AX, SEG INT_PROC
MOV DS, AX
LEA DX, INT_PROC
MOV AH, 25H
MOV AL, 45H
INT 21H
```

3. 请在下列空白处填空。下列程序段完成的功能是：输入一个字符串，统计其中字符‘a’的个数并存放在变量 X 中。

```
    |
    |
BUF DB 50, 51 DUP(0)
X   DB 0
    |
```

```
    LEA DX, BUF 或 MOV DX, OFFSET BUF
    MOV AH, 10
    INT 21H
    MOV BH, 0
    MOV BL, BUF+1
L1:  CMP BUF+1[BX], 'a'
    JNE LOP
    INC X
LOP: DEC BX
    JNZ L1
```


微助教上题目的解答：

汇编语言程序中的函数（子程序）定义伪指令与函数调用伪指令与 C 语言程序中的函数定义和调用语句非常相似。被调用函数与调用函数参数传递方法也是相似的，因为两种环境下的语句都要被编译成机器语言程序。试站在机器语言角度，解答如下问题。

(1)试以字节为单位画出刚进入子程序 CONVERT 时的堆栈示意图（假设①刚进入子程序时的(SP)=0100H；②BUF、L1 的偏移地址分别为 000AH，0256H）(提示：STDCALL 指明了函数定义中的参数按从左到右的顺序入栈)。

(2)在堆栈示意图中标出 NUM, RADIX, RESULT 的位置；

(3)在子程序 CONVERT 中访问 NUM 的寻址方式不同于传统的变量所对应的寻址方式，请问目标代码中应该是什么寻址方式。

子程序 CONVERT 定义及其调用语句如下：

```
CONVERT PROC NEAR STDCALL NUM:DWORD, RADIX:WORD,
RESULT:WORD
```

```
MOV EAX, NUM
```

```
MOV BX, RADIX
```

```
MOV SI, RESULT
```

```
.....
```

```
RET
```

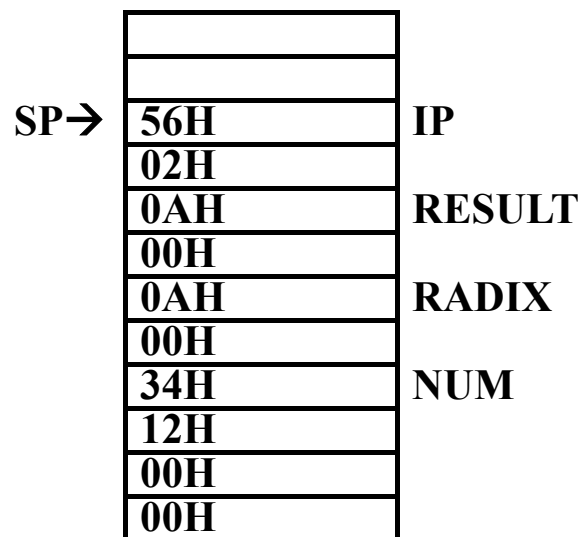
```
CONVERT ENDP
```

```
.....
```

```
INVOKE CONVERT, 1234H, 10, OFFSET BUF
```

```
L1:MOV AX, 10
```

```
.....
```



(3)变址寻址

编写实方式下的程序段，实现将中断矢量表中类型为 8 的中断处理程序的入口地址送到 OLDINT8 中保存的功能

CODE SEGMENT USE16

ASSUME CS:CODE, DS:CODE, SS:STACK

....

OLDINT8 DW ?, ?;保存原 INT 8H 的中断矢量

下面是你编写的程序段：

```
MOV    AX, 0  
MOV    DS, AX  
MOV BX, DS:[8H*4]  
MOV OLDINT8, BX  
MOV BX, DS:[8H*4+2]  
MOV OLDINT8+2, BX
```

```
MOV AH, 35H  
MOV AL, 8H  
INT 21H
```

结果： BX, ES

已知 BUF 存储区中存放着有符号数据，下列程序求 BUF 中的数据绝对值的和，计算结束后，结果保存到 RESULT 中。请在如下源程序的空白处填入适当的指令使程序功能完整。

BUF DB 30H, 89H, 0FEH

NUM EQU \$-BUF

RESULT DW ?

.....

MOV SI, OFFSET BUF 或 LEA SI, BUF

MOV CX, NUM

MOV DX, 0

AGAIN: MOV AL, [SI]

AND AL, AL

JGE NEXT1 ; JNS NEXT1

NEG AL

NEXT1: ADD DL, AL

JNC NEXT2

INC DH

NEXT2: INC SI

LOOP AGAIN

MOV RESULT, DX