《汇编语言程序设计》试卷（ A 卷参考答案）

2010-2011 学年第 2 学期计算机学院本科生
闭卷考试， 考试时间： 2011 年 7 月 2 日

专业_____ 班级_____ 学号_____ 姓名

题号	一	二	三	四	五	六	七	八	总分	核对人
题分	10	10	10	20	10	10	10	20	100	
得分										

得分	评卷人

一、填空题（共 10 分，每空 1 分）

1、 存储程序和程序控制是计算机工作的基本原理，要执行的程序预先存放在内存中 。CPU 当前将要执行的指令的物理地址依据 CS 和 IP/EIP 两个寄存器的内容计算得到。

2、CPU 在取到一条指令后，需要对该指令进行译码和执行，此时指令指示器(IP/EIP)会自动加上该指令的 代码长度(字节数) ，使其指向该指令的下一条指令。

3、在执行该指令时，若指令不是转移类（包括 条件 转移、无条件转移、子程序调用和返回、软中断调用和返回等）的指令时，就不会额外影响 CS 和 IP/EIP 的值。这样，取出的下一条指令就在该指令之下，程序将会顺序执行。

4、若该指令是段内子程序调用语句，CPU 首先将（IP/EIP） 压 栈 ， 然后将 子程序第一条指令的偏移地址 送入到 IP/EIP。这样取下一条指令时，取出的就是子程序中的第一条要执行的指令。

5、若该指令是段内子程序返回指令，CPU 将 栈顶元素弹出到 IP/EIP 。如果 CPU 在做上述操作前,栈顶的数据正好是在调用子程序时保存的断点偏移地址，则程序转回到了调用子程序的语句之下。

6、若该指令是 JMP WORD PTR[BX]，则该指令被称为无条件 段内间接 转移指令，转移到的目标指令的偏移地址存放在 DS 段中，通过 寄存器间接 寻址方式取到相应单元的值送给 IP 。

得分	评卷人

二、选择题（共 10 分，每题 1 分）

1、指令 ADD [BX], 20H 的错误原因是___D_____。
(A) BX 不能用于寄存器间接寻址方式　　(B) 源操作数不能用立即寻址方式
(C) 源、目的操作数不能同时为存储器操作数　(D) 源、目的操作数类型均不明确

2、指令 ADD [CX], AL 的错误原因是___A_____。
(A) CX 不能用于寄存器间接寻址方式　　(B) 源、目的操作数类型不匹配
(C) 源、目的操作数不能同时为存储器操作数　(D) 源、目的操作数类型均不明确

3、指令 ADD BX, AL 的错误原因是___B_____。
(A) BX, AL 的位置写反了　　(B) 源、目的操作数类型不匹配
(C) 源、目的操作数不能同时为存储器操作数　(D) 源、目的操作数类型均不明确

4、指令 MOV [DI], WORD PTR [SI] 的错误原因是___C_____。
(A) SI 不能用于寄存器间接寻址方式　　(B) DI 不能用于寄存器间接寻址方式
(C) 源、目的操作数不能同时为存储器操作数　(D) 目的操作数类型不明确

5、设 BUFW 为字变量， CON 为符号常量，下面四个语句中错误的语句是__B_____。
(A) MOV BUFW , AX　　(B) MOV CON , BUFW
(C) LEA AX , BUFW　　(D) MOV BUFW , CON

6、已知源操作数在堆栈段中，正确的指令是__D_____。
(A) MOV BYTE PTR [DI], [BP]　(B) MOV BYTE PTR [BP], '$'
(C) MOV SS:[SI], AX　　(D) MOV AX, SS:[SI]

7、 下列方法中不能访问一个外部设备寄存器的是 ___A_____。
(A) 使用 MOV 指令　　(B) 使用 IN / OUT 指令
(C) 使用 DOS 系统功能调用　　(D) 使用 BIOS 功能调用

8、 设（AX）=9011H， (BX)=2011H， 则执行 CMP AX，BX 后，有___C____。
(A) OF=0，SF=0　　(B) OF=0，SF=1
(C) OF=1，SF=0　　(D) OF=1，SF=1

9、设 (AX)=9011H，CF=0 则执行 ROL AX, 4 后，有___C_____。
(A) （AX）= 0119H， CF=0　　(B) (AX) = 1901H，CF=0
(C) （AX）= 0119H， CF=1　　(D) (AX) = 1901H，CF=1

10、设有 BUF DB 10，11 DUP(0)，现采用 10 号功能调用输入一个串到 BUF 缓冲区中。现要将实际输入串的长度送入 BX 中，正确的语句是___D_____。
(A) MOV BX, BUF+1　　(B) MOV BL, 10
(C) MOV BL, BUF+1　　(D) MOV BL, BUF+1
MOV BH, 0

得分	评卷人

三、简答题（共 10 分）

1．简要描述使用 TD.EXE 观察被调试程序数据段数据的操作方法。(5 分)

1、运行 TD.EXE，鼠标点击数据窗口；

2、选中该窗口，点击鼠标右键弹出命令列表；
3、选择 GOTO…；
4、在随之出现的窗口中输入：段首址：偏移量；
5、或者先在 CPU 窗口设置好 DS 寄存器，然后将”4”中输入：段首址：偏移量改为 DS:0，回车即见数据段开始的内容。

2．基于窗口的 WIN32 程序一般由哪四部分组成？窗口主程序完成的主要任务是什么？(5 分)

由主程序，窗口主程序，窗口消息处理程序和用户消息处理程序四部分组成。

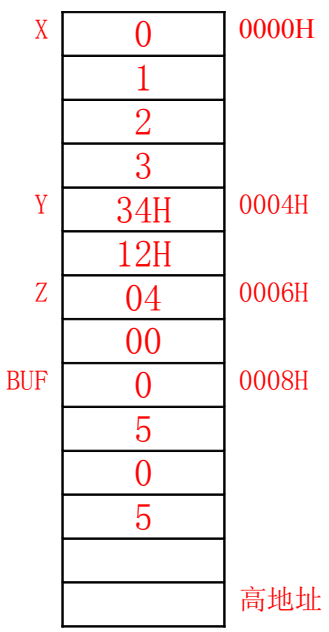
窗口主程序任务为：注册、创建、显示窗口，然后循环处理消息：获取、翻译、派发消息，当收到退出消息时，循环结束。

得分	评卷人

四、问答题（共 20 分）

1. 一个数据段定义如下：

```
DATA    SEGMENT  USE16
X       DB   0, 1, 2, 3
LEN  EQU   $ - X
Y       DW    1234H
Z     DW    Y
BUF   DB    2  DUP (0，5)
DATA    ENDS
```



（1）请在右表格中以字节为单位填写数据在存储器的存放形式，并标明 X、Y、Z、BUF 所处的偏移地址值。（7 分）

X、Y、Z、BUF 存储器内容正确 4 分
变量及偏移量标示正确 3 分

（2）执行如下指令后，寄存器的值是什么？（3 分）
每空 1 分

MOV CX, LEN　　则(CX)= 0004H

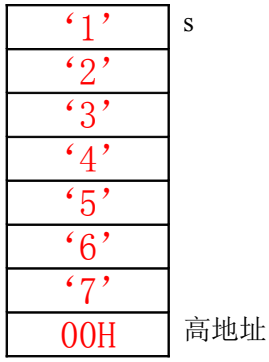
MOV BX, Z+1　　则(BX)= 0000H

MOV AX, Y[BX]　　则(AX)= 1234H

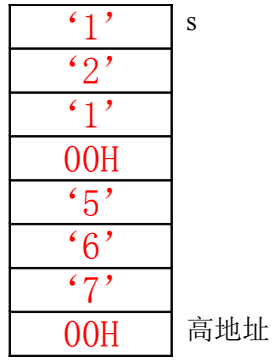
2、设有如下 C 语句：

```
char  s[8];
strcpy(s,"1234567");        /*将源串中各个字符的 ASCII 码顺序存放在 s 中*/
printf("%x",*(long *)(s+2));   /* L1 ，将(s+2)转换成双字类型地址，将其内容显示*/
*(short *)(s+2)=49;         /* L2 ， short 为字类型*/
printf("%s",s);             /* L3 */
```

(1) 以字节为单位画出执行 L1 处语句前和执行 L2 处语句后， 数组 s 的数据存放示意图。（4 分）



执行 L1 处语句前



执行 L2 处语句后

(2) 在执行 L1 处语句后，屏幕上看到的是 36353433。利用所学的汇编语言知识，解释产生该结果的原因。(3 分)

以（s+2）为地址取一个双字，由于双字中高字在高地址，低字在低地址，字中高字节在高地址，低字节在低地址，以十六进制显示即为 36353433H。

(3) 执行 L3 处语句，将在屏幕上显示什么？（3 分）

121

得分	评卷人

五、程序填空题（共 10 分，每空 1 分）

1. 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     BX, BUF ；或 MOV    BX, 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
```

2. 以下程序段的功能是将（BX）中的内容以十六进制串的形式显示到屏幕上，请在程序空白处填入空缺的代码。

```
F2T16   PROC
            PUSH    AX
            PUSH    CX
            PUSH    DX

            MOV     CX, 4
F2T16_L1:   ROL     BX, 4
            MOV     DX, BX
            AND     DX, 0FH
            CMP     DL, 9

            JBE     F2T16-L2 ；或 JLE  F2T16_L2
            ADD     DL, 7
F2T16_L2:   ADD     DL, 30H
            MOV     AH, 2
            INT     21H
            DEC     CX

            JNZ     F2T16-L1 ；或 JNE  F2T16_L1
            POP     DX
            POP     CX
            POP     AX

            RET
F2T16   ENDP
```

得分	评卷人

六、编写程序段（共 10 分）

1、试编写一个宏指令 W_SUB,实现字单元 X 与字单元 Y 中的内容相减后，差放入字单元 Z 中,即（X）-(Y)→ Z。要求调用宏指令前、后，相关寄存器中的内容保持不变。（5 分）

```
W_SUB  MACRO  X, Y, Z
       PUSH    AX
       MOV     AX, X
       SUB     AX, Y
       MOV     Z, AX
       POP     AX

       ENDM
```

2、编写一个程序段，将中断号为 21H 的中断处理程序入口处的偏移地址和段地址分别送入 AX 和 BX 中。（5 分）

```
MOV     AX, 0
MOV     DS, 0
MOV     AX, DS:[21H * 4]
MOV     BX, DS:[21H * 4 + 2]
```

得分	评卷人

七、分析程序（共 10 分）

1、已知 X, Y 是两个 short 类型（有符号字类型）的变量，试写出如下 C 程序段对应的汇编代码（3 分）。

注意：Statements1、Statements2 代表一些语句，在汇编代码中照抄即可。

```
if ( X >Y) {
      Statements 1
}
else {
      Statements 2
}
```

解答：MOV AX, X
CMP AX, Y
JLE L1

Statements1

JMP L2

L1: Statements2

L2:

2、如下的 C 语言程序段实现了一个 long 型（有符号双字类型）数组 a 中 10 个数据求和的功能，其编译后调试版本的汇编语言代码如下(注：为便于阅读，将有些具体的地址用符号代替了，黑色顶头部分为 C 语句，x,i 均是 long 类型的变量)。(7 分)

x=0;

```
        MOV   x, 0
```

for (i=0;i<10;i++)

```
        MOV   i, 0
        JMP   L1
L0:     MOV   EAX, i
        ADD   EAX, 1
        MOV   i,  EAX
L1:     CMP   i,  0Ah
        JGE   L2
```

x=x+a[i];

```
        MOV   ECX, i
        MOV   EDX, x
        ADD   EDX, a[ECX*4]
        MOV   x,  EDX
        JMP   L0
L2:
```

(1) 指出该段程序执行效率不高的原因 （2 分）。

答：没有给变量分配并固定使用某一个寄存器，例如，在将 i 加 1 时，先将 i 的值取出放到寄存器中，加 1 后，又送回到变量中，在下一次循环时，又是如此，若能固定使用一个寄存器可提高效率。此外对于 x 的计算，也应该在寄存器中进行，到计算结束再一次性将结果传送到变量 x 中。

关键：没有利用语句间的相关性进行优化。

(2)修改上述汇编代码段，以提高程序的执行效率(5 分)。ACM 班给出的程序中不允许出现 ADD 指令。

解答：

将 x 的值存入到 EDX 中(求和)，i 的值存入到 EAX 中（循环变量）

```
    MOV  EDX, 0
```

```
        MOV  EAX, 0
        JMP   L1
L0:  INC   EAX
L1:  CMP  EAX, 0AH
        JGE   L2
        ADD  EDX, a[EAX*4]
        JMP   L0
L2:  MOV  X, EDX
        MOV  i，EAX
```

ACM 班特殊部分(1 分)：用 LEA 指令、取补减法指令等替代 ADD。

得分	评卷人

八、程序设计（20 分）

设在以 BUF 为首址的字存储区中连续存储了多个有符号数(数据个数由编程者自己设定)，试编写一个完整的程序，将其中所有的正数皆以十六进制形式在屏幕上显示出来，不同数之间用逗号分开（但最后一个正数后面不加逗号），最后将正数的个数也以十六进制的形式在屏幕单独的一行上显示出来。

要求：

（1）子程序 F2T16（第五大题的第二小题）实现了一个字类型数据以十六进制形式显示的功能，要求调用该子程序完成相应的功能。在本程序设计中，对子程序的描述只需写出定义子程序 F2T16 的开始和结束伪指令即可，中间部分以省略号代替。ACM 班需要将该子程序改写成堆栈法传递参数，在子程序的描述中需要给出与之相关的代码。

（2） 画出主程序流程图 （2 分）。

（3） 写出变量、寄存器的使用分配情况和子程序入口和出口参数说明 （3 分）。

（4） 程序完整（包括数据段定义、堆栈段定义、代码段定义等）（15 分）。

解：

```
;  CX 用于循环次数控制，其值为要扫描的 BUF 缓冲区中字数据的个数
;  BX : 用于存放从缓冲区中读取的 一个字数据
;  SI :  缓冲区数据地址（指针）
; POSNUM ，用于存放正数的个数

.386
STACK  SEGMENT USE16 STACK
       DB 200 DUP(0)
STACK  ENDS
DATA   SEGMENT USE16
BUF     DW  33, -55, 0, 78, -20
NUM     EQU  ($-BUF)/2
POSNUM  DW  0
CRLF    DB   0DH,0AH,'$'
DATA   ENDS
CODE   SEGMENT USE16
          ASSUME DS:DATA,SS:STACK,CS:CODE
START: MOV  AX,DATA
       MOV    DS, AX
       MOV    CX, NUM
       MOV    SI, OFFSET  BUF
L1:               ;  下面采用循环方法，对缓冲区的各个数进行判断处理
       MOV     BX, [SI]
       CMP      BX, 0
       JLE       L3
       INC     POSNUM    ;  下面对正数进行处理，在第 1 个正数前面无需显示 逗号
       CMP    POSNUM,  1
       JE        L2
       MOV   DL, ','
       MOV   AH,  2
       INT    21H
L2:  CALL   F2T16
L3:  ADD    SI, 2
     LOOP   L1
     LEA    DX, CRLF
     MOV   AH,  9
     INT   21H
     MOV   BX, POSNUM
     CALL  F2T16
```

```
        MOV   AH,4CH
        INT   21H

; 以十六进制显示字数据
; 入口参数： BX，中存放待显示的数
; 出口参数：无
; 注：根据题目要求，此处可不抄
F2T16       PROC
            PUSH   AX
            。 。 。
            POP    AX
            RET
F2T16       ENDP
CODE    ENDS
        END   START
```

ACM 班特殊部分(2 分)：

1、主程序 CALL 之前，把

2、子程序开始要保护 BX

返回前恢复 BX，BP，返回前为 1？吗？ 2