汇编语言考试重点

必考内容：1.程序框架；

2.中断框架中保存向量或还原向量；

3.定义数据段；

4.宏：定义；比大比小。

试卷形式：一.十个选择（20 分）；

二.十个填空（20 分,包括三个程序填空）；

三.四个简答题（三个程序填空+一个写程序结果，1.数据段定义 2.中断（保存向量、代码放入中断中）、中断框架 3.宏）；

四.两道编程题（堆栈传参返回）。

各章节中重点：

第一章：补码；BCD 码（压缩和非压缩）；Ascall 码（30~39，大小写字母）； AX，CX（计数），SI, DI 的功能；DS,CS,SS,ES；8086 寻址方式；

第二章：重点指令： mov;

xhg,xlat;

lea,lds,les;

add,adc,inc;sub,sbb,dec,nec,cmp,nee,cmp;

乘除

cwd,cbw;

pushf,popf;

cld,std;

jmp 及其他条件转移指令；

ret,iret;

各类逻辑运算指令的功能（异或：某一位取反；与：某一位置零）；

test;

shl.shr;

flag：CF, OF, ZF;

loop，loopz，rep，repz；

CALL, RET（如何改变 sp 的值）；

第三章：程序框架；输入输出（课本 p68）；定位伪指令（课本 p74）；

变量定义；段定义；显示十进制的方法（除以 10，详见冒泡法实验）；

第四章：地址表；movs，scas，cmps；

第五章：宏定义；变量声明；输入输出（al/ax）；

复习方式：以 ppt 与实验为主

汇编语言模拟试题

模拟试题一

一． 选择题。（单项选择，30 分）

1. 下面说法正确的是（ ）
   A． 8086/8088 为 20 位 CPU。
   B． 不同的逻辑段的段地址有可能相同。
   C． 同一个逻辑段内所有单元的段地址相同，而偏移地址各不相同。
   D． SP 寄存器指示下一条要执行的指令的偏移地址。
2. Pentium CPU 有 32 根地址总线，内存的物理地址也为 32 位，则该 CPU 的寻址空间大小为（1G ＝ 1024M）（ ）
   A. 256M B. 16M C.1G D. 4G
3. 当 SS=9876H, SP=4328H，执行 POPF 指令后，堆栈顶部的物理地址是（ ）
   A. 9CA8AH B. 9CA90H C. 9CA86H D. 0DB00H
4. SAR(算术移位指令)可作一定程度上的除 2 运算，若 AL 中的内容为 93H，则指令 SAR AL，1 执行后，AL 中的内容为（ ）
   A．49H B. C9H C. 89H D. 92H
5. 已知（AL）＝41H，（AH）＝5AH，CPU 执行 SUB AL，AH 指令后，标志寄存器各位的值正确的是（ ）
   A. SF＝1，ZF＝0，CF＝1，OF＝0
   B. SF＝0，ZF＝0，CF＝1，OF＝0
   C. SF＝1，ZF＝0，CF＝0，OF＝1
   D. SF＝0，ZF＝1，CF＝0，OF＝1
6. 已知（AL）＝88H。如果该数是一个有符号数，它的十进制值为 X；如果它是一个压缩的 BCD 码，它的十进制值为 Y；执行 ADD AL,03H DAA 指令序列后，AL 的十六进制值为 Z。则（ ）
   A. X＝120，Y＝－120，Z＝91H C. X＝－120，Y=120，Z＝8BH
   B. X＝120，Y＝－88，Z＝8BH D. X＝－120，Y＝88，Z＝91H
7. 下列指令使用不正确的是（ ）
   A. SAL [100H], 1 C. MUL BYTE PTR 10H[BX+SI]
   B. AND BL, [BP+SI] D. MOV AL, －100
8. 下列指令使用正确的是（ ）
   A. MOV DS, IBABH C. MOV BYTE PTR[BX], BYTE PTR [SI]
   B. POP DL D. ADD BYTE PTR [100H], -1
9. 有符号数比较大小所用的跳转指令是（ ）
   A． JC 和 JE B. JS 和 JZ C. JA 和 JB D. JL 和 JG
10. 用一条指令实现将寄存器 BX 和 SI 的内容相加，结果并送入 AX 中，这条指令是（ ）
    A. ADD AX，[BX][SI] C. MOV AX, [BX][SI]
    B. AND AX, [BX][SI] D. LEA AX, [BX][SI]
11. 下面指令可能不能够将 AX 清零的是（ ）
    A．MOV AX，0 C. AND AX，0
    B．XOR AX，AX D. SBB AX,AX
12. 下面指令不能够将 AX 和 BX 内容调换的是（ ）
    A. XCHG AX, BX C. XCHG AL, BL

XCHG AH, BH

B. PUSH AX
PUSH BX
POP AX
POP BX

D. MOV CX, AX
MOV BX, CX
MOV AX, BX

13. 定义宏 Max，求两个数的最大数。定义如下：

```
Max Macro A,B,C
        CMP A,B
        JL  LESS
        MOV C,A
        JMP DONE
    LESS: MOV C,B
    DONE:
ENDM
```

则下列调用方式正确的是：（ ）

A. Max ax, bx, cx
B. CALL Max ax, 2, cx
C. Max 1,2,cx
D. Max ax,bl,cx

14. 8086CPU 的工作模式为 M1，Pentium CPU 复位后到引导 OS 之间工作模式为 M2，引导 OS 后正常工作模式为 M3，在 Window98 下运行一个 DOS 程序此时工作模式为 M4。则 M1，M2，M3，M4 为（ ）

A. 实模式，实模式，保护模式，保护模式
B. 实模式，保护模式，保护模式，虚拟 8086 模式
C. 虚拟 8086 模式，实模式，保护模式，保护模式
D. 实模式，实模式，保护模式，虚拟 8086 模式

15. DEBUG 调试程序中内存数据显示命令是（ ）

A. U 命令　　B. D 命令　　C. A 命令　　D. E 命令

二. 填空题。（20 分）

1. 在 C 语言中用“unsigned char a; signed short b;”定义两个变量 a,b，则 a 为无符号字节数，b 为 16 位补码数。写出 a，b 两个变量表示的十进制数的范围。a：_____，b：_____。
2. 已知 AX 寄存器的内容为 000FH，执行 DIV AL 后，AL 的值为_____。
3. 已知 AX 寄存器的内容为 FFFFH，执行 AND AH,81H 后，AH 的值为_____；执行 TEST AL,81H 后，AL 的值为_____。
4. 已知（BX）＝ 0005H，变量 Buffer 的偏移地址为 1000H，当前（DS）＝1000H。CPU 在执行 MOV AX，Buffer[BX]时，寻址的物理地址为_____。
5. 从 1234 号端口读入一个字节，并存放到 DL 寄存器的指令序列为：_____。
6. Li (i=1,2,…,7)为已定义的标号。现定义地址表 AddrTable：
   AddrTable DW L1,L2,L3,L4,L5,L6,L7。那么指令 JMP AddrTable[6]转移到的标号为：_____。
7. 两个模块中定义的 PUBLIC 段进行合并的条件是：_____相同且相同。

8. 模块 1 中定义了 Far 型的子程序 GlobalFunc，在模块 2 中要调用 GlobalFunc。为支持模块 1 和模块 2 通讯，模块 1 中使用的伪指令为：_____，模块 2 为：_____。

9. 指令序列：

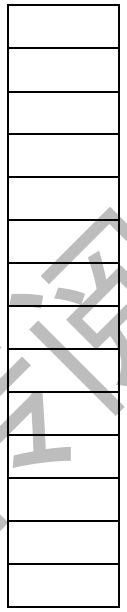
```
MOV AX, 0001H
MOV BX, 0002H
PUSH AX
PUSH BX
POP ECX
```

执行完后，32 位寄存器 ECX 的值为：_____。

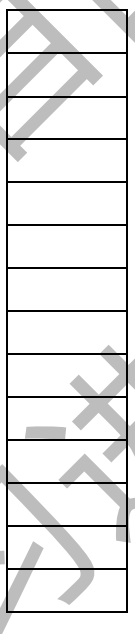
10. 根据下列数据定义伪指令，填写定义的数据在内存单元中的存放形式。

```
1．STRING   DB   ‘A’,’B’,’C’ ,’DEF’
            DW   ‘GH’
2．BUFFER   DW   1, -1
            DD   1234H, 2000H
3．DATA     DB   2   DUP(－1,2 DUP(1,2))
```

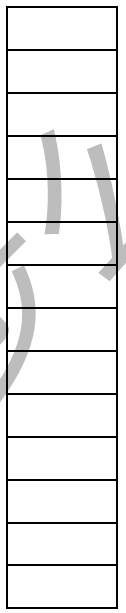
STRING



BUFFER



DATA



三． 分析下面程序段，回答指定问题。（11 分）

1．
```
MOV BL, 3EH
AND BL, 9AH
OR   BL, 78H
XOR BL, 56H
MOV CL,4
ROR BL,CL
```

问：执行上段程序后，BL=_____。

2．
```
ORG 1001H
DATA1 EQU   THIS BYTE
DATA2 DW    $,$+1,$+2,$+3
DATA3 DW    DATA1
```

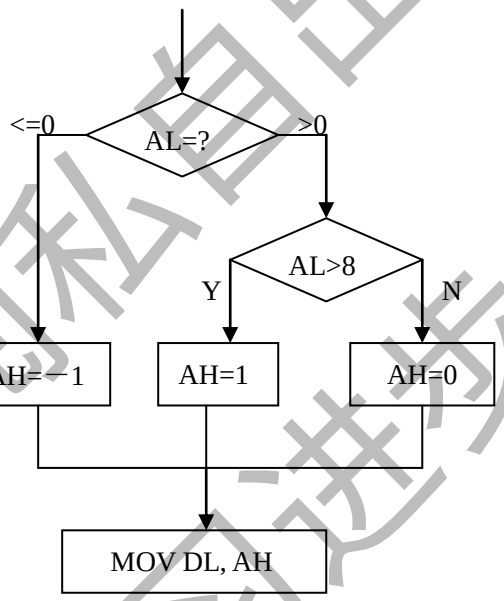
MOV AL,　DATA1
MOV BX,　DATA2+3
MOV CX,　DATA3
问：执行上段程序后，AL=_____，BX=_____，CX=_____。

3．SUB AX, AX
MOV BX, 1
MOV CX, 10
A：ADD AX, BX
ADD BX, 3
LOOP A
问：执行上段程序后，AX=_____，BX=_____，CX=_____。

四．编写完成下述功能的程序片段。（15 分）

1．在附加段中定义了一个字节缓冲区 Buffer，长度为 100。用串操作指令将这个缓冲区每个字节都置为 0FFH。

2．编写程序完成下列框图功能的程序片段。



3．在 C 语言中定义了一个字节型无符号数组 BUFFER，要求用__asm 嵌入汇编程序，来找出其中的最大值，送入变量 MAX 中。C 程序给出如下，编写嵌入的汇编代码。

```
int main()
{
    unsigned char BUFFER[100];   //定义字节型缓冲区
    unsigned char MAX;           //保存结果
    __asm{                       //嵌入汇编语言
    //在此处编写嵌入的汇编代码
```

```
    }
    return 0;
}
```

五． 程序设计。（编写完整的程序，24 分）

1．Buffer 缓冲区中有 100 个有符号 Word 数，查找中间是否存在数“-1”，如果存在，显示“Y”；否则，显示“N”。

2．数据区中定义了 4 个 Word 变量 X，Y，Z 和 Result，编写一个程序求 X、Y、Z 三数的平均值（不考虑余数），结果送入 Result 中。要求：求平均值部分写一个子程序 Average3；主程序和子程序之间所有参数都采用堆栈进行传递。

3．输入框（Edit）控件是 Windows 窗口程序常用的一种控件，它负责接受从键盘输入的字符。为控制 Edit 控件的外观特点，显示和对齐方式等，我们为它定义一些属性，每一个属性用一个 Word 的常数来表示，该数只有某一位（bit）为 1，其它位为 0。Edit 控件的所有属性用一个 Word 型的字来表示，该字的每一位（bit）表示一种相应属性，称该字为属性字。Edit 控件的属性定义如下：

```
ES_LEFT          EQU    0001H        ；左对齐
ES_CENTER        EQU    0002H        ；居中对齐
ES_RIGHT         EQU    0004H        ；右对齐
ES_MULTILINE     EQU    0008H        ；多行显示
ES_UPPERCASE     EQU    0010H        ；显示大写字母
ES_LOWERCASE     EQU    0020H        ；显示小写字母
ES_PASSWORD      EQU    0040H        ；密码框方式
ES_AUTOVSCROLL   EQU    0080H        ；自动加垂直滚动条
ES_AUTOHSCROLL   EQU    0100H        ；自动加水平滚动条
ES_READONLY      EQU    0200H        ；输入框只读
```

例如属性字 0009H 表示 Edit 控件属性为左对齐且多行显示；属性字 0202H 表示 Edit 控件属性为居中对齐且只读。

现假设 AX 寄存器里面存放要建立的 Edit 控件的属性字，该属性字包含一些初始属性。编写一个完整的程序，对该属性字进行处理，来按顺序完成下列功能：

1. 为 Edit 控件添加两个属性 ES_LEFT 和 ES_MULTILINE；
2. 测试 Edit 控件是否有属性 ES_RIGHT，如果有，删除该属性；
3. 测试 Edit 控件是否有属性 ES_AUTOVSCROLL，如果有，删除该属性；如果没有，添加该属性；
4. 测试 Edit 控件是否同时具有 ES_UPPERCASE 和 ES_LOWERCASE 两个属性。如果是，显示一个字符串“EditBox Style Error!”；否则，显示一个字符串“EditBox Style Right!”；
5. 程序结束。

程序如下：

```
;;;;;;;;;;;;;;;;  定义符号常数;;;;;;;;;;;;;;;;;;;;
    ES_LEFT                 EQU     0001H          ；左对齐
    ES_CENTER               EQU     0002H          ；居中对齐
```

```
ES_RIGHT        EQU     0004H       ；右对齐
ES_MULTILINE    EQU     0008H       ；多行显示
ES_UPPERCASE    EQU     0010H       ；显示大写字母
ES_LOWERCASE    EQU     0020H       ；显示小写字母
ES_PASSWORD     EQU     0040H       ；密码框方式
ES_AUTOVSCROLL  EQU     0080H       ；自动加垂直滚动条
ES_AUTOHSCROLL  EQU     0100H       ；自动加水平滚动条
ES_READONLY     EQU     0200H       ；输入框只读
;;;;;;;;;;;;;;;  你的代码如下;;;;;;;;;;;;;;;;;;
```

模拟试题一参考答案

一.　　选择题（15*2 =30）

CDABA　DADDD　DDADB

二.　　填空（1-9，每题 1’；　10 题 7’, 2+3+2）

1. 0-255,　-32768-32767
2. 1
3. 81H,0FFH
4. 11005H
5. mov dx,1234　in al,dx　　mov dl,al
6. L4
7. 段名　类型属性
8. public globalfunc　　extern globalfunc:FAR
9. 00010002H
10.

STRING	
	‘A’
	‘B’
	‘C’
	‘D’
	‘E’
	‘F’
	‘H’
	‘G’

BUFFER	
	01H
	00H
	FFH
	FFH
	34H
	12H
	00H
	00H
	00H
	20H
	00H
	00H

DATA	
	FFH
	01H
	02H
	01H
	02H
	FFH
	01H
	02H
	01H
	02H

三.　　分析下面程序段，回答指定问题(11’)

1. 0C2H　(2’)
2. 01H,0710H,1001H(3’)

3. 145,31,0 (6’)

四. 编写完成下述功能的程序片段

1. (5’)

```
CLD                 (1’)
MOV CX,100          (1’)
MOV AL,0FFH         (1’)
LEA DI,BUFFER       (1’)
REP STOSB           (1’)
```

2. (5’)

```
    CMP AL,0
    JG NEXT         (1’)
    MOV AH,-1
    JMP DONE        (1’)
NEXT:
    CMP AL,8
    JG NEXT2        (1’)
    MOV AH,0
    JMP DONE        (1’)
NEXT2:
    MOV AH,1        (1’)
DONE:
    MOV DL,AH
```

3. (5’)

```
    MOV AL,BUFFER
    MOV ECX,100             (1’)
    MOV ESI,0               (1’)
AGAIN:
    CMP AL,BUFFER[ESI]
    JAE NEXT                (1’)
    MOV AL,BUFFER[ESI]
NEXT:
    INC ESI                 (1’)
    LOOP AGAIN
    MOV MAX,AL              (1’)
```

五. 程序设计。

1. (6’)

```
DATA SEGMENT                    (1’)
    BUFFER DW 100 DUP(?)        (1’)
DATA ENDS
CODE SEGMENT
START:
```

```
        ASSUME DS:DATA,CS:CODE
        MOV AX,DATA
        MOV DS,AX

        MOV CX,100
        MOV SI,0
AGAIN:
        CMP BUFFER[SI],-1
        JE FOUND            (1')
        ADD SI,2            (2')
        LOOP AGAIN
        MOV AH,2
        MOV DL,'N'
        INT 21H
        JMP DONE
FOUND:
        MOV AH,2            (1')
        MOV DL,'Y'
        INT 21H
DONE:
        MOV AH,4CH
        INT 21H
CODE ENDS
        END START
```

2. (9')

```
DATA SEGMENT            (1')
        X DW ?          (1')
        Y DW ?
        Z DW ?
        RESULT DW ?
DATA ENDS
CODE SEGMENT
        ASSUME DS:DATA,CS:CODE
START:
        MOV AX,DATA
        MOV DS,AX
        SUB SP,2        (1')
        PUSH X          (1')
        PUSH Y
        PUSH Z
        CALL AVERAGE3
        POP RESULT      (1')
        MOV AH,4CH
```

```
        INT 21H
AVERAGE3 PROC NEAR    (1')
        PUSH BP
        MOV BP,SP

        PUSH AX
        PUSH DX
        PUSH BX
        MOV AX, 4[BP]    (1')
        ADD AX, 6[BP]
        ADD AX, 8[BP]
        CWD
        MOV BX,3
        IDIV BX          (1')
        MOV 10[BP], AX

        POP BX
        POP DX
        POP AX
        POP BP
        RET 6      (1')
AVERAGE3 ENDP
CODE ENDS
        END START
```

3.(9')

```
DATA SEGMEMT       (1')
        STRING1 DB 'EditBox Style Error$' (1')
        STRING2 DB 'EditBox Style Right$'
DATA ENDS
CODE SEGMENT
        ASSUME DS:DATA,CS:CODE
START:
        MOV AX,DATA
        MOV DS,AX
        OR AX, ES_LEFT       (1')
        OR AX, ES_MULTILINE

        TEST AX, ES_RIGHT   (1')
        JZ GOON
        XOR AX, ES_RIGHT
GOON:                        (2')
        XOR AX, ES_AUTOVSCROLL
```

```
        TEST AX, ES_UPPERCASE    (2')
        JNZ RIGHT
        TEST AX, ES_LOWERCASE
        JNZ RIGHT
        PUSH AX
        MOV AH,9                 (1')
        LEA DX, STRING1
        INT 21H
        POP AX
        JMP DONE
RIGHT:
        PUSH AX
        MOV AH,9
        LEA DX, STRING2
        INT 21H
        POP AX
DONE:
        MOV AH,4CH
        INT 21H
CODE ENDS
        END START
```

模拟试题二

一． 选择题。（单项选择，30 分）

1. 用来存放下一条将要执行的指令代码段地址的寄存器是（　　）。
   A. SP　　B. ES　　C. IP　　D. CS
2. 要使串处理指令从高地址向低地址顺序连续执行，应先执行的指令是（　　）。
   A. STI　　B. CLC　　C. STD　　D. CLD
3. 已知（AL）＝41H，（AH）＝5AH，CPU 执行 SUB AL，AH 指令后，标志寄存器各位的值正确的是（　　）
   A. SF＝1，ZF＝0，CF＝1，OF＝0
   B. SF＝0，ZF＝0，CF＝1，OF＝0
   C. SF＝1，ZF＝0，CF＝0，OF＝1
   D. SF＝0，ZF＝1，CF＝0，OF＝1
4. 使状态标志位 CF 清零的错误指令是（　　）。
   A. SUB AX，AX　　B. CLC　　C. OR AX，AX　　D. MOV CF，0
5. 下面的数据传送指令中，错误的操作是（　　）。
   A. MOV WORD PTR[BX],1000H　　C. OUT 200, AL
   B. MOV CX,1000H　　D. MOV [BP+DI],1000H
6. SAR(算术移位指令)可作一定程度上的除 2 运算，若 AL 中的内容为-3，则指令 SAR AL，1 执行后，AL 中补码数的值为（　　）
   A. -3　　B. -1.5　　C. -1　　D. -2

7.宏 MyMacro 定义如下：

```
MyMacro Macro A,B,C
            CMP A,B
            JL LESS
            MOV C,A
            JMP DONE
    LESS:   MOV C,B
    DONE:
ENDM
```

则下列调用方式正确的是：（　　）

A. MyMacro [1000H], bx, cx
B. CALL MyMacro ax, 2, cx
C. MyMacro [1000H], [1002H], cx
D. MyMacro ax, bl, 1

8.执行下列指令后，SP 寄存器的值正确的是（　　）。

```
MOV   SP，  2000H
PUSH  BX
```

A. 1FFFH　　B. 1FFEH　　C. 2001H　　D. 2002H

9.完成 BL+CL→AL 的错误操作是（　　）。

A.
```
ADD AL，BL
ADD AL，CL
```

B.
```
ADD BL，CL
MOV AL，BL
```

C.
```
XCHG AL，CL
ADD AL，BL
```

D.
```
MOV AL，BL
ADD AL，CL
```

10. 执行下面指令序列后,正确的结果是（　　）。

```
MOV AH，40H
ADD AH，AH
ADC AH，AH
ADC AH，AH
```

A. AH=00H　　B. AH=100H　　C. AH=200H　　D. AH=01H

11. 执行下列程序段后，正确的结果是（　　）。

```
MOV AL，80H
MOV BL，08H
CBW
IDIV BL
```

A. AH=00H　　B. AH=0FFH　　C. AH=00H　　D. AH=0FFH
AL=10H　　AL=0F0H　　AL=0F0H　　AL=10H

12. 完成当 BX≥0 时转 GREAT，当 BX<0 时转 LITTLE，错误的操作是（　　）。

A.
```
OR BX，BX
JS LITTLE
GREAT:
…
```

B.
```
TEST BX，8000H
JZ GREAT
```

C.
```
CMP BX，0
JGE GREAT
LITTLE:
…
```

D.
```
MOV AX，BX
JS LITTLE
```

LITTLE: GREAT:

… …

13. 定义如下地址表：ADDRTABLE DW L1,L2,L3,L4,L5,L6，则指令 JMP ADDRTABLE[4]转入的标号是（ ）

A. L2 B. L3 C.L4 D.L5

14. 下面指令组完成将字单元 BUF1 和 BUF2 的内容互换，错误的操作是（ ）。

A. XCHG BUF1，BUF2

B. MOV AX，BUF1
MOV BX，BUF2
MOV BUF2，AX
MOV BUF1，BX

C. MOV AX，BUF1
XCHG AX，BUF2
MOV BUF1，AX

D. MOV AX，BUF1
MOV BX，BUF2
XCHG AX，BX
MOV BUF1，AX
MOV BUF2，BX

15. 两个模块中分别定义两个逻辑段 S1,S2，下列不是 S1、S2 合并的必要条件的是（ ）

A. S1、S2 的组合属性必须都为 PUBLIC 或者 STACK
B. S1、S2 的对齐属性必须相同
C. S1、S2 的类别必须相同
D. S1、S2 的段名必须相同

二. 填空题。（20 分）

1. 为了方便用户使用外设，IBM－PC 机提供了两种典型的例行子程序供用户编程调用，它们是_____和_____系统功能调用。它们都是系统编制的子程序，通过_____方式来调用所需的子程序。
2. 在算术运算中，判断无符号数运算是否溢出应根据标志位_____ ，判断有符号数运算是否溢出应根据标志位_____。
3. 8086/8088 系统中，一个 20 位的物理地址是由_____和_____组成的。已知（BX）＝0008H，（SI）=0002H，变量 Buffer 的偏移地址为 1000H，当前（DS）＝1000H。CPU 在执行 XCHG AX，Buffer[BX][SI]时，寻址的物理地址为_____。
4. 80386 系统中，(EAX)=00001234H, (EBX)=00000002H, 则操作数 4[EAX][EBX*4]的地址为：_____。
5. 模块 1 中定义了 WORD 型的变量 BUFFER，在模块 2 中要直接使用 BUFFER。为支持模块 1 和模块 2 通讯，模块 1 中使用的伪指令为：_____，模块 2 为：_____。
6. 写出对应指令，完成如下功能。
   a) 将 AX 的低四位清零，其它不变。_____
   b) 将 BX 的低四位置 1，其它不变。_____
   c) 将 CX 的低四位求反，其它不变。_____
7. 用数据定义伪指令定义数据块 BLOCK，要求数据依次为：3 个 BYTE 型的－1，5 个 WORD 型的 0，7 个 DWORD 型的 1。

_____

8. 设有宏定义如下：

```
SHIFT   MACRO   X，Y
        MOV CL，X
        SAL Y，CL
ENDM
```

请将宏调用语句：SHIFT 5，BX 进行宏展开后的指令序列为：_____
_____。

9. 8088/8086 系统中，8 号定时器中断的中断向量的偏移地址和段地址存放在内存中的物理地址分别为_____和_____。

10. DEBUG 调试环境中，反汇编命令是：_____；单步执行命令是_____；内存查看命令是_____。

三. 分析下面程序段，回答指定问题。（15 分）

1.
```
MOV AL，BL
NOT AL
ADD AL，BL
```
问：该程序段执行后：AL= _____，CF=_____。

2.
```
MOV   AL，10H
MOV   BL，AL
SHL   AL，1
SHL   AL，1
ADD   AL，BL
SHL   AL，1
```
问：① 该程序段完成的功能是 _____。
②该程序段执行后：AL=_____，BL=_____。

3. 设 AX=1234H，BX=3456H，CX=5678H
执行：
```
MOV   CH，AH
MOV   CL，AL
XCHG BX，CX
MOV   AH，CH
MOV   AL，CL
```
问：① 该程序段完成的功能是_____。
② AX=_____，BX= _____，CX= _____。

四. 编写完成下述功能的程序片段。（15 分）

1. 编写一中断服务子程序 ISR，从 100 号端口读入一个字节，取反后输出到 101 号端口。

2. 在数据段中定义了一个字节缓冲区 AA，附加段中定义了个字节缓冲区 BB，长度都为 100。用 MOVSB 指令将 AA 中的数据全部复制到 BB。

3. 在 C 语言中定义了一个字节型数组 DATA，要求用__asm 嵌入汇编程序，来找出其中正数的个数，送入字节变量 nPos 中。C 程序给出如下，编写嵌入的汇编代码。

```
int main()
{
    char DATA[100] = {-1,0,1,2,-2,……};   //定义字节型缓冲区
```

```
    char nPos = 0;                          //保存结果
    __asm{                                  //嵌入汇编语言

        //在此处编写嵌入的汇编代码

    }
    return 0;
}
```

五． 程序设计。（编写完整的程序，20 分）

1．SCORE 缓冲区中有 100 个无符号 WORD 数，求它们的平均值，送入 AVERAGE 字单元。

2．DATA 缓冲区存放有 10 个无符号字节型数据， RESULT 缓冲区存放有 10 个无符号字型数据。现定义函数 $Y=f(X)=X^2+X-1$。编程完成将 DATA 中的数依次按 f(X)计算，并将结果分别送入 RESULT 中。要求：计算 f(X)部分用子程序 CalcF 实现，并且传入传出参数全部通过堆栈进行传递。

模拟试题二参考答案

一． 选择题（15*2 =30）

DCADD　DABAD　CDBAB

二． 填空（每题 2’）

1. DOS, BIOS，中断
2. CF OF
3. 段地址 偏移地址　1100AH
4. 00001240H
5. public BUFFER　extern/extrn BUFFER: word
6.
   a) AND AX，0FFF0H
   b) OR　BX，000FH
   c) XOR CX，000FH
7. BLOCK　DB 3 DUP(-1)
   DW 5 DUP(0)
   DD 7 DUP(1)
8. MOV CL，5
   SAL　BX，CL
9. 00020H　　00022H
10. U　T　D

三． 分析下面程序段，回答指定问题 (每题 5')

1． AL= 0FFH，CF= 0。

2． ① 该程序段完成的功能是 AL×10，
② AL= 0A0H 或 160 ，BL= 10H

3． ① 该程序段完成的功能是 AX 和 BX 的内容互换。
② AX= 3456H ，BX= 1234H ，CX= 3456H

四． 编写完成下述功能的程序片段

1. (5')

```
ISR   PROC   FAR
      STI
      PUSH AX
      IN AL, 100
      NOT AL
      OUT 101,AL
      POP AX
      IRET
 ISR ENDP
```

2. (5')

```
      MOV CX,100
      CLD
      LEA SI,AA
      LEA DI,BB
      REP MOVSB
```

3. (5')

```
      MOV CX,100
      MOV ESI,0
      MOV AL,0
AGAIN:
      CMP DATA[ESI],0
      JLE/JNG NEGATIVE
      INC AL
NEGATIVE:
      INC ESI
      LOOP AGAIN
```

五． 程序设计。

1. (10')

```
DATA SEGMENT                    (程序框架 2')
      SCORE   DW 100 DUP(?)
DATA ENDS
CODE SEGMENT
      ASSUME DS:DATA,CS:CODE
```

```
START:
        MOV AX,DATA
        MOV DS,AX
        MOV CX,100
        MOV SI,0
AGAIN:
        ADD AX, SCORE[SI]
        ADC DX,0
        ADD SI,2
        LOOP AGAIN

        MOV BX,100
        DIV   BX
        MOV AVERAGE, AX

        MOV AH,4CH
        INT 21H
CODE ENDS
        END START
```

2. (10’)

```
DATASEG SEGMENT
        DATA DB 10   DUP(1)
        RESULT DW 10 DUP(?)
DATASEG ENDS
CODE SEGMENT
        ASSUME DS:DATASEG,CS:CODE
START:
        MOV AX,DATASEG
        MOV DS,AX
        MOV CX,10
        MOV SI,0
        MOV DI,0
AGAIN:
        SUB SP,2
        MOV AL,DATA[SI]
        PUSH AX
        CALL CalcF
        POP RESULT[DI]
        INC SI
        ADD DI,2
        LOOP AGAIN
        MOV AH,4CH
        INT 21H
CalcF PROC NEAR
```

```
        PUSH BP
        MOV BP,SP
        PUSH AX
        PUSH BX
        MOV AX, 4[BP]
        MOV BL,AL
        MUL AL
        ADD AL,BL
        ADC AH,0
        SUB AX,1
        MOV 6[BP], AX
        POP BX
        POP AX
        POP BP
        RET 2
CalcF ENDP
CODE ENDS
        END START
```

模拟试题三

一． 选择题。（单项选择，30 分）

1. 下面说法正确的是（ ）
   A. 8086/8088 为 20 位 CPU。
   B. 不同的逻辑段的段地址有可能相同。
   C. 同一个逻辑段内所有单元的段地址相同，而偏移地址各不相同。
   D. SP 寄存器指示下一条要执行的指令的偏移地址。

2. 下列寄存器是 16 位的是（ ）。
   A. IF　　B. SP　　C. EIP　　D. BL

3. 已知（AL）＝41H，（AH）＝5AH，CPU 执行 SUB AL，AH 指令后，标志寄存器各位的值正确的是（ ）
   A. SF＝1，ZF＝0，CF＝1，OF＝0
   B. SF＝0，ZF＝0，CF＝1，OF＝0
   C. SF＝1，ZF＝0，CF＝0，OF＝1
   D. SF＝0，ZF＝1，CF＝0，OF＝1

4. 完成将 CX 寄存器清零，并且使进位标志 CF 置零，错误的指令是（ ）。
   A. SUB CX，CX　　C. XOR CX，CX
   B. MOV CX，0000H　　D. AND CX，0000H

5. 下面的数据传送指令中，错误的操作是（ ）。
   A. MOV DS,AX　　C. XCHG BX,[BX]
   B. MOV DX,[1000H]　　D. MOV [AX],1000H

6. 已知（AL）＝88H。如果该数是一个有符号数，它的十进制值为 X；如果它是一个压缩的 BCD 码，它的十进制值为 Y，则（ ）
   A. X＝120，Y＝－120　　C. X＝－120，Y=120

B. X＝120，Y＝－88　　D. X＝－120，Y＝88

7.宏 ADD3 定义如下：

```
ADD3  Macro  A, B, C
          ADD A,B
          ADD A,C
 ENDM
```

先要实现 1＋2＋3，则下列代码正确的是：（　　）

A. ADD3　1, 2, 3

B.
```
MOV AL,1
MOV BL,2
CALL ADD3 AL,BL,3
```

C.
```
MOV AL,1
ADD3 AL, 2, 3
```

D.
```
MOV AL,2
ADD3 1,AL, 3
```

8.执行下列指令后，SP 寄存器的值正确的是（　　）。

```
MOV   SP，  2000H
PUSH  BX
```

A. 1FFFH　　B. 1FFEH　　C. 2001H　　D. 2002H

9.下列指令的源操作数的段地址在 DS 中的是（　　）。

A. MOV AX, [BP][DI]　　C. MOV AX,4[BX][SI]

B. MOV AX, SS:2[BX]　　D. MOV AX, ES:[8*4]

10. 有符号数比较大小所用的跳转指令是（　）

A. JC 和 JE　　B. JS 和 JZ　　C. JA 和 JB　　D. JL 和 JG

11. 定义如下地址表：ADDRTABLE　DW　L1,L2,L3,L4,L5,L6，则如需要转入 L3，下列代码正确的是（　　）

A. JMP ADDRTABLE[3]　　C. JMP ADDRTABLE[5]

B. JMP ADDRTABLE[4]　　D. JMP ADDRTABLE[6]

12. 下面指令不能够将 AX 和 BX 内容调换的是（ ）

A.
```
XCHG AX, BX
```

C.
```
XCHG AL, BL
XCHG AH, BH
```

B.
```
PUSH AX
PUSH BX
POP AX
POP BX
```

D.
```
MOV CX, AX
MOV BX, CX
MOV AX, BX
```

13. 完成对寄存器 DX 的无符号数乘以 4 的正确操作是（　　）

A.
```
SHL DX，1
SHL DX，1
```

C.
```
ROL DX，1
ROL DX，1
```

B.
```
MOV CL，4
SHL DX，CL
```

D.
```
MOV CL，2
RCL DX，CL
```

14. 若 AL=96H，BL=01H，分别执行 MUL BL 和 IMUL BL 指令后，结果正确的是（　　）。

A. AX=0096H　B. AX=0096H　C. AX= 0FF96H　D. AX=150
AX=0096H　AX= 0FF96H　AX= 0096H　AX=-150

15. DEBUG 调试程序中内存数据显示命令是（　）
A. U 命令　B.　D 命令　C. A 命令　D. E 命令

二. 填空题。（20 分）

1. 8088/8086 系统采用内存分段技术，逻辑段长度不得超过_____。
2. 十进制数 3 和–3 的 8 位补码数分别为：_____、_____。
3. 在 C 语言中用“unsigned char a; signed short b;”定义两个变量 a,b，则 a 为无符号字节数，b 为 16 位补码数。写出 a，b 两个变量表示的十进制数的范围。
a：_____，b：_____。
4. 下面的伪指令定义后，

```
ORG　2000H
BUF1　DB　10　DUP　(?)
BUF2　DW　$, -1
```

则 LEA AX, BUF2 执行后，AX 的值为：_____，MOV AX, BUF2 执行后，AX 的值为：_____。
5. 两个模块中定义的 PUBLIC 段进行合并的条件是：_____相同且_____相同。
6. 模块 1 中定义了 Far 型的子程序 GlobalFunc，在模块 2 中要调用 GlobalFunc。为支持模块 1 和模块 2 通讯，模块 1 中使用的指令为：_____，模块 2 为：_____。
7. 编写程序实现：
a. 将 AL 与 DX 中的两个无符号数相加，结果放入 DX 中。
_____
b. 将 AL 与 DX 中的两个有符号数相加，结果放入 DX 中。
_____
8. 从 200H 号端口读入一个字节，并存放到 DL 寄存器的指令序列为：
_____。
9. 8086 系统中，将中断服务子程序 MyTimer 挂接到系统的 1CH 号中断上的程序代码为_____。
10. 指令序列：

```
MOV EAX, 00010002H
PUSH EAX
POP AX
POP BX
```

执行完后，AX 和 BX 的分别值为：_____、_____。

三. 分析下面程序段，回答指定问题。（15 分）

1.

```
MOV SI, 0
MOV DI,0
CLD
MOVSW
```

LODSB
STOSB

问：程序段执行后：SI=_____ ，DI= _____。

2. MOV AX，0FFFH
MOV BX，0FFF0H
XOR AX，BX
XOR BX，AX
XOR AX，BX

问：程序段执行后：AX=_____ ，BX= _____。

3. SUB AX, AX
MOV BX, 1
MOV CX, 8
A：ADD AX, BX
SHL BX, 1
LOOP A

问：执行上段程序后，AX=_____，BX=_____，CX=_____。

四． 编写完成下述功能的程序片段。（15 分）

1. 编写伪指令，定义一个数据段 DATA，并在 DATA 中定义数据块 BLOCK，要求数据依次为：一个字符串’Assembly’，30 个 BYTE 型的 1，50 个 WORD 型的 0，70 个 DWORD 型的-1。
2. 编写一个宏 Max，求 3 个无符号 WORD 型立即数的最大值，并存入 AX。
3. 在 C 语言中定义了一个字节型数组 DATA，要求用__asm 嵌入汇编程序，将数组中的每一个数最高位清 0，最低位置 1，其他位保持不变。C 程序给出如下，编写嵌入的汇编代码。

```
int main()
{
  char DATA[100] = {1,0,1,2,3,……};  //定义字节型缓冲区
  __asm                              //嵌入汇编语言
  {
     //在此处编写嵌入的汇编代码
```

```
  }
  return 0;
}
```

五． 程序设计。（编写完整的程序，20 分）

1. 以 Block 为首地址的内存中有 100 个有符号 WORD 数，编写一个程序统计这 100 个数中有多少个正数，并将结果送到 Result 字节单元中。
2. C 函数 tolower 完成的功能是将一个大写字母转换为小写字母，如果该字符是小写字母或者其他字符时不作转换；而函数 toupper 则将一个小写字母转换为大写字母，如果该字符是大写字母或者其他字符时不作转换。要求：

i. 用汇编语言编写两个子程序 tolower 和 toupper 来实现上述功能。要求：传入参数为原字符，传出参数为转换过的字符，所有参数都通过堆栈传递；
ii. 调用上述子程序将字符串“HELLO Assembly!”分别转化为全部大写字母和全部小写字母的字符串，并打印到屏幕。

模拟试题三参考答案

一. 选择题（每题 2’）

CBABD DCBCD BDABB

二. 填空（每题 2’）

1. 64K
2. 03H (0000,0011B) 0FDH(1111,1101B)
3. 0~255 (0~2^8-1) -32768~32767(-2^{15} ~ 2^{15}-1)
4. 200AH 200AH
5. 段名 类别属性
6. public GlobalFunc extern/extrn GlobalFunc: far
7. a) mov ah,0 add dx, ax
   b) cbw add dx, ax
8. mov dx,200h in al,dx mov dl,al
9. MOV ES,0
   MOV ES:[1CH*4], offset MyTimer
   MOV ES:[1CH*4+2], SEG MyTimer
   或者
   (MOV AX, SEG MyTimer
   MOV DS, AX
   MOV DX, OFFSET MyTimer
   MOV AL, 1CH
   MOV AH, 25H
   INT 21H)
10. 0002 0001

三. 分析下面程序段，回答指定问题 (4，4，7)

1. 0003H，0003H
2. 0FFF0H 0FFFH
3. 00FFH(255) 0100H(256) 0000H

四. 编写完成下述功能的程序片段

1. (5’)

```
DATA SEGMENT                          (1’)
    BLOCK   DB ‘Assembly’             (1’)
            DB 30 DUP(1)              (1’)
            DW 50 DUP(0)              (1’)
            DD 70 DUP(-1)             (1’)
```

```
DATA ENDS
```

2. (5’)

```
Max MACRO X,Y,Z                 (1’)
 LOCAL GOON, DONE               (1’)
        MOV AX, X               (1’)
        CMP AX,Y                (1’)
        JAE GOON
        MOV AX,Y
GOON:
        CMP AX,Z                (1’)
        JAE DONE
        MOV AX,Z
DONE:
ENDM
```

3. (5’)

```
        MOV ECX, 100                        (1’)
        MOV EBX, 0                          (1’)
AGAIN:
        AND DATA[EBX], 7FH/0111,1111B       (1’)
        OR DATA[EBX], 1                     (1’)
        INC EBX                             (1’)
        LOOP AGAIN
```

五. 程序设计。

1. (8’)

```
DATA SEGMENT                    (程序框架 2’)
        BLOCK DW 100 DUP(?)     (2’)
        RESULT DB 0
DATA ENDS
CODE SEGMENT
START:
        ASSUME DS:DATA,CS:CODE
        MOV AX,DATA
        MOV DS,AX
        MOV CX,100
        MOV SI,0
AGAIN:
        CMP BLOCK[SI], 0
        JLE GOON                (2’)
        INC RESULT
GOON:
        ADD SI, 2               (2’)
        LOOP AGAIN
        MOV AH,4CH
```

```
        INT 21H
CODE ENDS
        END START
```

2. (12’)

```
DATASEG SEGMENT
        STR DB 'HELLO Assembly!'
        STR1 DB 20 DUP('$')
        STR2 DB 20 DUP('$')
DATASEG ENDS
CODE SEGMENT
        ASSUME DS:DATASEG,CS:CODE
START:
        MOV AX,DATASEG
        MOV DS,AX
        MOV CX,15
        MOV SI,0
AGAIN:
        MOV AL,STR[SI]
        MOV AH,0
        SUB SP,2                (1’)
        PUSH AX                 (1’)
        CALL tolower
        POP AX                  (1’)
        MOV STR1[SI],AL
        INC SI
        LOOP AGAIN
        MOV AH,9                (1’)
        LEA DX,STR1
        INT 21H
        MOV CX,15
        MOV SI,0
CONTINUE:
        MOV AL,STR[SI]
        MOV AH,0
        SUB SP,2
        PUSH AX
        CALL toupper
        POP AX
        MOV STR2[SI],AL
        INC SI
        LOOP CONTINUE
        MOV AH,9
        LEA DX,STR2
```

```
        INT 21H
    MOV AH,4CH
        INT 21H

tolower PROC NEAR
            PUSH BP
            MOV BP,SP
            PUSH AX
            MOV AX,4[BP]    (1')

            CMP AL,'A'      (1')
            JB DONE
            CMP AL,'Z'
            JA DONE
            ADD AL,20H
    DONE:
            MOV 6[BP],AX    (1')
            POP AX
            POP BP
            RET 2           (1')
tolower ENDP

toupper PROC NEAR
            PUSH BP
            MOV BP,SP
            PUSH AX
            MOV AX,4[BP]    (1')

            CMP AL,'a'      (1')
            JB DONE1
            CMP AL,'z'
            JA DONE1
            SUB AL,20H
    DONE1:
            MOV 6[BP],AX    (1')
            POP AX
            POP BP
            RET 2           (1')
toupper ENDP
CODE ENDS
            END START
```