

汇编语言 期中考试

类型：独立完成开卷考试，不得参考他人（包括往届）答案

截止日期：本学期最后一周周日（含）

上交方式：打印，手写，扫描为pdf后交给助教，文件格式“期中-姓名-学号.pdf”

姓名：陈红瑞

学号：202200130053

诚信声明：我承诺本开卷考试由本人独立手写扫描完成，没有参考他人的答案。
(在下面抄写)

我承诺本开卷考试由本人独立手写扫描完成，没有参考他人的答案。

一、单项选择题 (10x2 = 20)

1. 将机器语言目标文件组装成可执行文件的实用程序是 (C)。
A. 解释程序 B. 编译程序
C. 链接程序 D. 汇编程序
2. 设 SP 初值为 1000H，执行指令 CALL FAR PTR PROC 后，SP 为 (C)。
A. 0FFFH B. 1002H
C. 0FFEh D. 0FFCH
3. 能保存当前执行指令的下一条指令的地址的寄存器称为 (A)。
A. 指令指针寄存器 B. 数据寄存器
C. 段寄存器 D. 地址寄存器
4. 完成将 AL 和 CF 清零不能使用的指令是 (D)。
A. SUB AL, AL B. XOR AX, AX
C. AND AL, 0D D. MOV AL, 0_{CF=1}
5. 下列指令中错误的是 (D)。
A. MOV WORD PTR SS:[BX], 2008H
B. ADD DS:[BP+DI], 0800H
C. MOV DX, 2008H
D. MOV WORD PTR [SP], 08H
6. MOV AL, 85H
SUB AL, 0AFH
上述指令执行后，标志位 CF 和 OF 的值是 (A)。

A. CF=1, OF=0
C. CF=1, OF=1

B. CF=0, OF=0
D. CF=0, OF=1

7. 如果 AL 中仅有存放的数据的第 3 位 (最低为第 0 位) 为 1, 则转到标号 LAB1 去执行。不能实现上述转移的程序段是 (A)

A. MOV CL, 4
SHL AL, CL
JC LAB1
AND AL, 08H
JNZ LAB1

B. MOV CL, 4
SHR AL, CL
JC LAB1
TEST AL, 08H
JNZ LAB1

8. 对大端字节序的计算机而言, 设物理地址 (10FF0H)=10H, (10FF1H)=20H, (10FF2H)=30H, 如从地址 10FF1H 中取出一个字的内容是 (A)

A. 2030H B. 2010H C. 1020H D. 3020H

9. 已知 DS=1000H, SS=2000H, BP=2000H, DI=1234H, 则指令 MOV AX, [BP+DI+2] 的源操作数据在 (D) 中。

A. 物理地址 23234H
B. 物理地址 13234H
C. 物理地址 23236H
D. 物理地址 13236H

10. 下面指令语句中, 语法正确的是 (C)。

A. MOV BYTE PTR [BX], 5000H
B. JMP SHORT PTR OPR
C. CMP WORD PTR [BX], 60
D. INC [BX]

二、填空题 (10x3 = 30)

1. 对于 8086 而言, 假设某中断的中断号为 12H, 则其中断向量的 CS 位于物理地址 0120H 48H

2. 如果当前 (DS)=2100H, (BP)=0158H, (DI)=10A5H, 则执行指令 LEA BX, [BP+DI] 后, (BX) = 21F0H

3. MOV AX, SS:[BX+SI+3] 的寻址方式是 相对基址变址寻址

4. 将 36 进制数 F6U5 转换为 10 进制是 708701。

5. (+25) 的 8 位 16 进制补码表示是 19H, (-38) 的 8 位 16 进制补码表示是 DAH。

6. 外设和主机交换 (传送) 数据的四种基本方式是 直接交换、轮询、中断、成组。

三、名词解释 (5x4 = 20)

1. 汇编语言
由助记符和操作数作为组成的汇编码作为基本元素的、直接描述计算机硬件基本操作序列的编程语言。

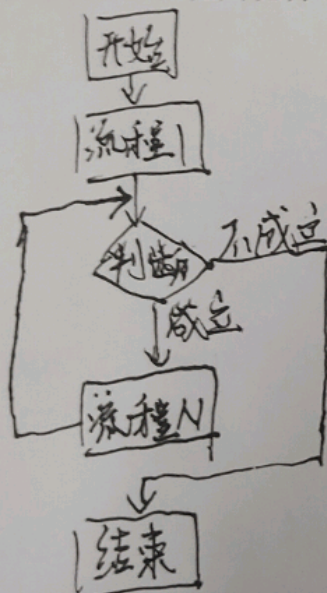
2. 8086 的指令分类

- ① 数据传送指令。 ② 算术运算指令。 ③ 位操作指令
④ 串操作指令 ⑤ 控制转移指令 ⑥ 处理器控制指令

3. 微机系统

计算机系统中由系统软件和硬件组成的部分, 包括主板固件, 操作系统, 微处理器, 存储器, I/O 设备。

4. 画流程图说明“当...成立”型循环的循环结构



四、程序阅读 (10)

阅读下列程序, 回答问题:

- (1) 该程序 (很可能) 实现什么功能?
(2) 已知该程序中有两个 (类) 错误, 它们分别是什么错误?
(3) 如果 (2) 中的错误都被修正, 程序执行完毕后, ARRAY 的值是多少?

	TITLE	WHATSTHIS
LEN	EQU	10


```

DATA    SEGMENT
ARRAY   DW      524,14,44,111,8831,451,741,891,981,423
DATA    ENDS

CODE    SEGMENT
        ASSUME  CS:CODE, DS:DATA
START:
        MOV     AX, DATA
        MOV     DS, AX
        XOR     SI, SI

OUTER:
        XOR     DI, DI

INNER:
        MOV     AX, ARRAY[DI]
        MOV     BX, ARRAY[DI+1]
        CMP     AX, BX
        JB      SKIP
        MOV     ARRAY[DI], BX
        MOV     ARRAY[DI+1], AX

SKIP:
        ADD     DI, 2
        CMP     DI, LEN*2
        JNZ     INNER
        ADD     SI, 1
        CMP     SI, LEN
        JNZ     OUTER
        MOV     AH, 4CH
        INT     21H

CODE    ENDS

        END     START

```

(1) 实现的功能为将数据段中的 ARRAY 数组中的数据
进行两两比较, 并将较小的数放在左边。这里执行
两层循环, 外层循环一共执行 10 次, 内层循环负责依次将相邻
两个数比较并将小的数放在低地址, 因此可能实现的是冒泡
排序

(2) ① ~~在外层循环中, XOR DI, DI 应改为 MOV SI, DI,~~
~~MOV DI, SI,~~ ① 将 ADD DI, 2 改为 ADD DI, 1, 否则无法实现正确
的比较过程。
② CMP DI, LEN*2 改为 CMP DI, LEN, 否则会将 ARRAY 数组外的数
参与比较 (3) ARRAY 的值为 14, 44, 111, 423, 451, 524, 741, 891, 981, 8831

计 60~69 分, 70~79 分, 80~89 分, 90~100 分段 (均含) 的整数平均分并
别存放在 A6, A7, A8, A9 和 A10 存储单元中, 其中负责统计单个分数段
的部分要求使用过程或宏实现, 并在主程序中针对各个区间分别调用该过程
或宏。退出程序可使用 4CH 号调用。
参考程序开头:

	TITLE	AVGSCORE
STUNUM	EQU	113
DATA	SEGMENT	
GRADE	DB	76,69,84,90,73,88,99,63,100,80,
A6	DB	0
A7	DB	0
A8	DB	0
A9	DB	0
A10	DB	0
...		
DATA	ENDS	

```

COMP MACRO LO, HIHI, SCORE, VAL
; 用宏实现比较分数段
    CMP LO, SCORE
    JG EXIT ; 不在分数段内, 且比分数段低
    CMP HI, SCORE
    JL EXIT ; 比分数段高
    INC VAL ; 在分数段内, 将对应的值加 1
EXIT: ADD VAL, 0

```

ENDM

CODE SEGMENT

ASSUME CS:CODE, DS:DATA

```

START: MOV AX, DATA
        MOV DS, AX
        MOV CX, STUNUM ; 设置循环次数
        MOV BX, 0
        XOR DL, DL ; 寄存器初始化
        LOOP1:
            MOV DL, 60
            MOV DH, 69

```


COMP DL, DH, ~~SCORE~~ GRADE[BX], A6 ; ~~在60-69~~ 在60-69

MOV DL, 70

MOV DH, 79

COMP DL, DH, GRADE[BX], A7 ; 在70-79, A7+1

MOV DL, 80

MOV DH, 89

COMP DL, DH, GRADE[BX], A8 ; 在80-89时 A8+1

MOV DL, 90

MOV DH, 100

COMP DL, DH, GRADE[BX], A9 ; 在90-100时 A9+1

INC BX

LOOP LOOP1 ; 执行循环, BX也加1

MOV AX, 4CH

INT 21H

; 退出程序

CODE ENDS

END

START