第5章 汇编语言程序设计 习题与解答

> 收录范围：ch5_v2023 课件全部课堂例题（p59-108）、课堂习题（p109-116）、教材习题答案页（p126-132）及课件附带的第4章习题答案页（p117-125），共原题 45 道；另按本章考点补充 7 道，合计 52 道。课件只给答案未给题干的（教材习题），按答案还原题意并注明；课件代码疑似笔误处标注「存疑」并给出修正版。所有程序为 8086/MASM 语法，可汇编。

一、课件原题

（一）课堂例题

题1（课件 p59-60，例5-35）

利用 DOS 系统功能调用实现简单人机对话：屏幕显示提问 "WHAT IS YOU NAME? "，然后接收键盘输入的姓名字符串，最后返回 DOS。写出完整程序。

解答：

```
DATA    SEGMENT
BUF     DB   100              ; 缓冲区第1字节：容量100
        DB   ?                ; 第2字节：预留，系统回填实际字符数
        DB   100 DUP(?)       ; 第3字节起：存放键入字符串(0DH结束)
MSG     DB   'WHAT IS YOU NAME? $'   ; 9号功能显示串，'$'结束
DATA    ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA
START:  MOV  AX,DATA          ; 取数据段段基址
        MOV  DS,AX            ; 装填DS（ASSUME不装填）
DISP:   MOV  DX,OFFSET MSG    ; DX=提问串偏移地址
        MOV  AH,09            ; 9号功能：显示字符串
        INT  21H              ; 屏幕显示提问
KELBD:  MOV  DX,OFFSET BUF    ; DX=输入缓冲区首址
        MOV  AH,0AH           ; 0AH号功能：键盘输入字符串
        INT  21H              ; 等待键入，回车结束
        MOV  AH,4CH           ; 4CH号功能：返回DOS
        INT  21H
CODE    ENDS
        END  START            ; 入口START
```

执行过程：先显示提问串（遇 '$' 停），再等待用户键入姓名并按回车，字符串存入 BUF+2 起、实际长度写入 BUF+1，最后返回 DOS。考点：9 号与 0AH 号缓冲区的区别（'$' 结尾 vs 三段结构）。

题2（课件 p63-64）

多项式 $f(X)=5X^3+4X^2-3X+2$，编制程序计算 X=5 时的函数值，结果存 RESULT 单元。

解答： 分析（秦九韶法）：$f(X)=((5X+4)X-3)X+2$，只需 3 次乘法。

```
DATA    SEGMENT
X       DW   5                ; 自变量
RESULT  DW   ?                ; 结果单元
DATA    ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA
START:  MOV  AX,DATA          ; 装填数据段
        MOV  DS,AX
        MOV  AX,5             ; 系数5
        MUL  X                ; AX=5*X=25 (DX:AX,DX=0)
        ADD  AX,4             ; AX=5X+4=29
        MUL  X                ; AX=(5X+4)X=145
        SUB  AX,3             ; AX=(5X+4)X-3=142
        MUL  X                ; AX=((5X+4)X-3)X=710
        ADD  AX,2             ; AX=712
        MOV  RESULT,AX        ; 保存结果
        MOV  AH,4CH           ; 返回DOS
        INT  21H
CODE    ENDS
        END  START
```

中间结果逐步：AX=5→25→29→145→142→710→712。最终 RESULT=712=02C8H。验算：5×125+4×25-3×5+2=625+100-15+2=712 ✓。注意 MUL 为字乘，结果在 DX:AX，本题数值小 DX=0。

题3（课件 p65-66）

内存自 TABLE 起七个单元连续存放 0~6 的立方值，任一数 X($0\le X\le 6$) 在 FIRST 单元中。查表求 X 的立方值存入 SECOND 单元。给出两种方法。

解答:

```
DATA    SEGMENT
TABLE   DB   0,1,8,27,64,125,216   ; 立方表
FIRST   DB   3                ; X（课件写作DB X，X应为0~6常数，此处取3示例）
SECOND  DB   ?                ; 结果
DATA    ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA
START:  MOV  AX,DATA
        MOV  DS,AX
; 方法一：地址计算法
        MOV  BX,OFFSET TABLE  ; BX=表首址
        MOV  AH,0             ; AH清0，AX=X
        MOV  AL,FIRST         ; AL=X
        ADD  BX,AX            ; BX=TABLE+X
        MOV  AL,[BX]          ; 取表项
        MOV  SECOND,AL        ; 存结果
; 方法二：XLAT查表指令（替换方法一中间四句）
;       MOV  BX,OFFSET TABLE
;       MOV  AL,FIRST
;       XLAT                  ; AL←DS:[BX+AL]
;       MOV  SECOND,AL
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

X=3 时 SECOND=27=1BH。XLAT 以 BX 为表首址、AL 为索引取 [BX+AL] 内容送 AL；课件注释 "(AL+BX)→AL" 应理解为取该地址单元的内容。课件原文 `FIRST DB X` 中 X 未定义，属简写（存疑：X 需另有 EQU 定义或为立即数）。

题4（课件 p67-69）

A、B 单元各有一个 8 位无符号数（设不相等），比较大小并在屏幕显示 "A>B" 或 "B>A"。画流程图并编程。

解答：流程图：开始 → A→AL，比较 B 和 AL → 判断 B>A? 是→'B'→DL、'A'→BL；否→'A'→DL、'B'→BL → 依次显示 DL、'>'、BL → 结束（课件 p68 图，见笔记 §10）。

```
DATA    SEGMENT
A       DB   20H              ; 无符号数32
B       DB   92H              ; 无符号数146
DATA    ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA
START:  MOV  AX,DATA
        MOV  DS,AX
        MOV  AL,A             ; AL=A
        CMP  B,AL             ; B-AL，置标志
        JA   BGA              ; 无符号B>A则转BGA
        MOV  DL,'A'           ; A>B：大者'A'先显示
        MOV  BL,'B'           ; 小者'B'后显示
        JMP  DISP
BGA:    MOV  DL,'B'           ; B>A：大者'B'
        MOV  BL,'A'
DISP:   MOV  AH,2             ; 2号功能显示DL
        INT  21H
        MOV  DL,'>'           ; 显示'>'
        INT  21H
        MOV  DL,BL            ; 显示小者
        INT  21H
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

执行结果：92H>20H，JA 成立，显示 **B>A**。考点：无符号比较用 JA；CMP 内存,寄存器 的方向；显示顺序用 DL/BL 分工避免重复代码。

题5（课件 p71）

求 10 个带符号数中的最大数。M1 起存 10 个字节带符号数，结果存 M2 单元。

解答：

```
DATA    SEGMENT
M1      DB   21H,2AH,31H,0BAH,0,1,2,3,4,5   ; 10个带符号数(BAH=-70)
M2      DB   1 DUP(?)       ; 结果单元
DATA    ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA
START:  MOV  AX,DATA
        MOV  DS,AX
        MOV  SI,OFFSET M1   ; 源指针
        MOV  DI,OFFSET M2   ; 结果指针
        MOV  CX,0009H       ; 比较次数=10-1=9
        MOV  AL,[SI]        ; 先取第1个数作当前最大
DDD:    CMP  AL,[SI+1]      ; 与下一个数比较
        JG   NEXT           ; 带符号：AL大则跳过
        MOV  AL,[SI+1]      ; 否则更新最大值
NEXT:   INC  SI             ; 指针后移
        DEC  CX             ; 次数减1
        JNZ  DDD            ; 未完继续
        MOV  [DI],AL        ; 存最大数
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

逐行模拟：AL=21H；依次与 2AH(更新)、31H(更新)、BAH(负数不更新)、0,1,2,3,4,5(均不更新) 比较，共 9 次；最终 M2=31H。考点：带符号用 JG；n 个数比 n-1 次；DEC CX+JNZ 与 LOOP 等价。

题6（课件 p74）

把 0、1、2、…、9 输出在屏幕上（计数法循环）。

解答：

```
CODE    SEGMENT
        ASSUME CS:CODE
START:  MOV  CX,0AH          ; 循环10次
        MOV  BL,30H          ; '0'的ASCII，初始化
DDD:    MOV  DL,BL           ; 待显示字符
        MOV  AH,02H          ; 2号功能
        INT  21H             ; 显示（循环体）
        INC  BL              ; 下一数字ASCII
        DEC  CX              ; 计数减1
        JNZ  DDD             ; 循环控制
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

执行结果：屏幕显示 0123456789。BL 从 30H 增到 39H，CX 从 10 减到 0 退出。本例无数据段，故无需装填 DS。

题7（课件 p75-77）

BUF 起存 20 个 8 位带符号数，统计负数个数存 NEGA 单元并显示（设个数 0~9）。计数法。

解答：

```
DATA    SEGMENT
BUF     DB   2,-1,56,-38,4,7,-9,0,12,-1,3,5,-6,8,10,-2,14,16,-18,20
COUNT   EQU  20              ; 数据个数
NEGA    DB   ?               ; 负数个数
DATA    ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA
START:  MOV  AX,DATA
        MOV  DS,AX
        MOV  SI,OFFSET BUF   ; 源指针
        MOV  DL,0            ; 负数计数器清0
        MOV  CX,COUNT        ; 循环20次（初始化）
LP:     MOV  AL,[SI]         ; 取数
        AND  AL,AL           ; 自身相与，置SF等标志
        JNS  NEXT            ; SF=0(非负)跳过
        INC  DL              ; 为负，计数+1（循环体）
NEXT:   INC  SI              ; 指针后移
        LOOP LP              ; CX-1非零继续（控制）
        MOV  NEGA,DL         ; 存结果
        OR   DL,30H          ; 个数转ASCII(0~9)
        MOV  AH,2
        INT  21H             ; 显示
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

示例数据中负数为 -1,-38,-9,-1,-6,-2,-18 共 7 个 → NEGA=7，屏幕显示 '7'。考点：AND AL,AL 不改变值只置标志；JNS 判符号位；OR DL,30H 等价 ADD DL,30H（0~9 时）。

题8（课件 p78-79）

STR 起存放以 '$' 结束的字符串（长度≤100），统计长度存 LEN 单元。条件控制法。

解答:

```
DATA    SEGMENT
STR     DB   'ABCDJKLLHH$'  ; 以'$'结束
LEN     DB   0              ; 长度初值0
DATA    ENDS
STACK   SEGMENT PARA STACK 'STACK'
STT     DB   100 DUP(?)     ; 堆栈区
STACK   ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA,SS:STACK
START:  MOV  AX,DATA
        MOV  DS,AX
        MOV  AX,STACK       ; 装填堆栈段
        MOV  SS,AX
        LEA  DI,STR         ; DI=串首址
        MOV  AL,'$'         ; AL=结束标志（初始化）
LP:     CMP  AL,[DI]        ; 当前字符是'$'吗
        JZ   STOP           ; 是则退出（控制）
        INC  LEN            ; 否：长度+1（循环体）
        INC  DI             ; 指针后移
        JMP  LP             ; 继续
STOP:   MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

示例串 'ABCDJKLLHH' 共 10 字符 → LEN=10。考点：次数未知用条件控制（CMP+JZ），不用 CX；LEN 为内存单元可直接 INC。

题9（课件 p80-83）

冒泡排序：TABLE 起一串 8 位有符号数，长度在 LEN（EQU $-TABLE），从小到大排序。写出算法与程序。

解答： 算法（课件 p80）：① 外循环次数→CH；② 表首址→SI；③ 内循环次数→CL；④ 交换标志 BL 清 0；⑤ 相邻比较顺序正确转⑦；⑥ 否则交换并置 BL=1；⑦ SI+1；⑧ CL-1 非零转⑤；⑨ BL=0 已排好转结束；⑩ CH-1 非零转②。

```
DATA    SEGMENT
TABLE   DB   12,78,-2,0,75,15
LEN     EQU  $-TABLE        ; =6
DATA    ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA
START:  MOV  AX,DATA
        MOV  DS,AX
        MOV  CH,LEN         ; CH=6
        DEC  CH             ; 外循环上限=n-1=5
LOP0:   MOV  SI,OFFSET TABLE; 每趟从表头开始
        MOV  CL,CH          ; 内循环次数=当前趟数
        XOR  BL,BL          ; 交换标志清0
LOP1:   MOV  AL,[SI]        ; 取相邻前者
        CMP  AL,[SI+1]      ; 与后者比较
        JLE  NEXT           ; 带符号：前者<=后者顺序对
        XCHG AL,[SI+1]      ; 交换：AL⇔后者
        MOV  [SI],AL        ; 完成交换
        OR   BL,01H         ; 置交换标志
NEXT:   INC  SI             ; 下一对
        DEC  CL
        JNZ  LOP1           ; 内循环未完继续
        AND  BL,BL          ; 本趟有交换吗
        JZ   STOP           ; 无交换→已有序，提前结束
        DEC  CH
        JNZ  LOP0           ; 外循环未完继续
STOP:   MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

跟踪：第1趟后 12,-2,0,75,15,78；第2趟后 -2,0,12,15,75,78；第3趟无交换 BL=0 提前退出。结果 -2,0,12,15,75,78。考点：JLE 带符号；XCHG AL,[SI+1] 一条指令完成"读后者+写前者暂存"；交换标志提前终止。

题10（课件 p84-85）

冒泡排序：把 M1 开始的 100 个带符号数从小到大排序，仍放 M1 开始的单元（从后往前扫描的优化版）。

解答：

```
DATA    SEGMENT
M1      DB   100 DUP(?)     ; 100个带符号数（课件记X1..X100）
DATA    ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA
START:  MOV  AX,DATA
        MOV  DS,AX
        MOV  BL,02          ; 内循环下界初值2
NEXT3:  MOV  SI,OFFSET M1+99; 每趟从表尾开始
        MOV  CH,64H         ; 交换标志=100(未交换)
        MOV  CL,64H         ; 内循环计数初值100
NEXT2:  MOV  AL,[SI]        ; 取后者
        CMP  AL,[SI-1]      ; 与前者比较
        JGE  NEXT1          ; 带符号：后者>=前者顺序对
        XCHG AL,[SI-1]      ; 交换
        MOV  [SI],AL
        MOV  CH,CL          ; 记录本趟发生过交换
NEXT1:  DEC  SI             ; 向前扫
        DEC  CL
        CMP  CL,BL
        JGE  NEXT2          ; CL>=BL继续内循环
        CMP  CH,64H         ; 本趟无交换？
        JZ   DONE           ; 是→已有序结束
        INC  BL             ; 下界上移1(最小者已就位)
        JMP  NEXT3          ; 下一趟
DONE:   MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

执行分析：第1趟 CL 从 100 减到 2，比较 (99,98)…(1,0) 共 99 对，最小者浮到 M1[0]；第2趟下界 BL=3，比较到 (2,1)，次小者就位；每趟下界加 1。若某趟 CH 保持 64H（无交换）则提前结束。与题9对比：扫描方向相反、用 BL 收缩下界代替 CH 递减趟数。课件双栏排版，左栏末 `NEXT1:DEC SI` 接右栏首 `DEC CL`（已按图 p85 还原）。

题11（课件 p87-89，例5-42）

将存储区中两个单字节数相加，结果送第 3 字节，并转换成十进制数送 CRT 显示（设结果<100）。用子程序实现。

解答：

```
DATA    SEGMENT
LIST    DB   16,48,?          ; 两加数与结果单元
DATA    ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA
START:  MOV  AX,DATA
        MOV  DS,AX
        CALL ADSUB           ; 调用求和子程序
        CALL OUTPR           ; 调用显示子程序
        MOV  AH,4CH
        INT  21H
ADSUB   PROC                 ; 求和：内存单元传参
        MOV  AL,LIST         ; 取第1数16
        ADD  AL,LIST+1       ; +第2数48 → 64
        MOV  LIST+2,AL       ; 存第3字节
        RET                  ; 返回
ADSUB   ENDP
DISPL   PROC                 ; 显示DL中一位十进制数
        OR   DL,30H          ; 转ASCII
        MOV  AH,02
        INT  21H
        RET                  ; 存疑：课件p88漏写此RET，必须补上
DISPL   ENDP
OUTPR   PROC                 ; 拆分十位个位并显示
        MOV  AL,LIST+2       ; 取和64
        MOV  AH,0            ; AX=64
        MOV  BL,10
        DIV  BL              ; AL=商6(十位)，AH=余4(个位)
        MOV  DL,AL           ; 十位送DL
        MOV  BH,AH           ; 个位暂存BH
        CALL DISPL           ; 显示十位'6'
        MOV  DL,BH
        CALL DISPL           ; 显示个位'4'
        RET
OUTPR   ENDP
CODE    ENDS
        END  START
```

执行结果：屏幕显示 **64**。考点：DIV BL 商 AL 余 AH；子程序嵌套调用（OUTPR 调 DISPL）；参数用内存单元 LIST 传递。**存疑**：课件 DISPL 过程体无 RET，照抄会顺序滑入 OUTPR 形成无限递归直至堆栈溢出；上面已补 RET。

题12（课件 p90-92，5.6.3例1）

两个 10 字节压缩 BCD 数分别存于 A、B 起的数据区（低位在低地址），相加结果存 C 起数据区。

解答：

```
DATA    SEGMENT
A       DB   12H,38H,63H,0,0,0,0,0,0,0   ; 被加数(示例补齐10字节)
B       DB   88H,27H,42H,0,0,0,0,0,0,0   ; 加数
C       DB   11 DUP(?)      ; 结果10字节+1进位字节
N       EQU  10             ; 字节数
DATA    ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA
START:  MOV  AX,DATA
        MOV  DS,AX
        MOV  SI,OFFSET A    ; 被加数指针
        MOV  DI,OFFSET B    ; 加数指针
        MOV  BX,OFFSET C    ; 结果指针
        MOV  CX,N           ; 循环10次
        CLC                 ; 清进位（仅一次）
NEXT:   MOV  AL,[SI]        ; 取被加数字节
        ADC  AL,[DI]        ; 带进位加加数字节
        DAA                 ; 压缩BCD加法调整
        MOV  [BX],AL        ; 存本位和
        INC  SI
        INC  DI
        INC  BX             ; 三指针后移
        LOOP NEXT           ; 循环控制
        MOV  AL,0
        ADC  AL,0           ; 收最高进位
        MOV  [BX],AL        ; 存第11字节
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

首字节模拟：12H+88H=9AH → DAA：低半 A>9 加 06H、高半 A>9 加 60H → AL=00H、CF=1（12+88=100 ✓）；次字节 38H+27H+CF1=60H（低半 8+7+1=16 产生半进位 AF=1）→ DAA 加 06H → 66H ✓（38+27+1=66）。考点：ADC+DAA 进位链；CLC 只清一次；结果区多留 1 字节存进位。减法改 SBB+DAS 同理。

题13（课件 p93，练习）

把 M1(A) 起 4 个单元 ASCII 码十进制数与 M2(B) 起 4 个单元 ASCII 码十进制数相加，和放 M3(C) 起单元。课件只给分析，补全程序。

解答： 课件分析：① 屏蔽高 4 位（35H→05H）；② 用非组合加法调整指令 AAA；③ 数据段如下。补全代码：

```
DATA    SEGMENT
A       DB   31H,32H,33H,34H   ; '1234'，低位在低地址→数4321
B       DB   35H,36H,37H,38H   ; '5678'→数8765
C       DB   5 DUP(?)          ; 和(非压缩BCD，5位)
DATA    ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA
START:  MOV  AX,DATA
        MOV  DS,AX
        MOV  SI,OFFSET A
        MOV  DI,OFFSET B
        MOV  BX,OFFSET C
        MOV  CX,4              ; 4位数字
        CLC                    ; 清进位
LP:     MOV  AL,[SI]
        AND  AL,0FH            ; ASCII→非压缩BCD(屏蔽高4位)
        MOV  AH,[DI]
        AND  AH,0FH            ; 加数同样屏蔽
        ADC  AL,AH             ; 带进位相加
        AAA                    ; 非压缩加法调整，进位入CF
        MOV  [BX],AL           ; 存本位(非压缩BCD)
        INC  SI
        INC  DI
        INC  BX
        LOOP LP
        MOV  AL,0
        ADC  AL,0              ; 收最高进位
        MOV  [BX],AL
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

跟踪：位0：1+5=6；位1：2+6=8；位2：3+7=10→AAA→AL=0,CF=1；位3：4+8+1=13→AAA→AL=3,CF=1；末进位 1。C=06,08,00,03,01 → 4321+8765=13086 ✓。若要求和也为 ASCII，存前 `OR AL,30H` 即可。

题14（课件 p94-96，例2）

A 起存多位非压缩十进制数（低位在低地址），B 中一位非压缩十进制数，求乘积存 C 起存储区。

解答：

```
DATA    SEGMENT
A       DB   7,5,3,6        ; 数6357(低位在低地址)
N       EQU  $-A            ; =4
B       DB   8              ; 一位乘数
C       DB   5 DUP(?)       ; 积最多5位
DATA    ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA
START:  MOV  AX,DATA
        MOV  DS,AX
        MOV  SI,OFFSET A    ; 被乘数指针
        MOV  DI,OFFSET C    ; 积指针
        MOV  BYTE PTR [DI],0; 积首字节清0
        MOV  CX,N           ; 4轮
        CLC
RETRY:  MOV  AL,[SI]        ; 取一位被乘数
        INC  SI
        MUL  B              ; AX=该位×8
        AAM                 ; 乘法调整：AH=十位,AL=个位
        ADD  AL,[DI]        ; 积低位+上一轮进位(高位)
        AAA                 ; 加法调整(若AL>9则AH+1)
        MOV  [DI],AL        ; 存本位
        INC  DI
        MOV  [DI],AH        ; 存进位到下一字节
        LOOP RETRY
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

逐轮跟踪：① 7×8=56→AAM→AH5,AL6；+0→6 存 C0，C1=5。② 5×8=40→AH4,AL0；+5=5 存 C1，C2=4。③ 3×8=24→AH2,AL4；+4=8 存 C2，C3=2。④ 6×8=48→AH4,AL8；+2=10→AAA→AL0,AH5；存 C3=0，C4=5。C=06,05,08,00,05 → 50856=6357×8 ✓。考点：AAM 拆十位个位；AAA 处理与上轮高位的相加；进位逐字节传递。

题15（课件 p97-98，例3）

BUF 起三个字节非压缩 BCD 码（低位在低地址）转换为二进制数，结果存 DX。

解答：

```
DATA    SEGMENT
BUF     DB   2,5,6          ; 数652
M1      DB   100            ; 百位权
M2      DB   10             ; 十位权
DATA    ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA
START:  MOV  AX,DATA
        MOV  DS,AX
        LEA  BX,BUF         ; BX=表首址
        MOV  DH,0
        MOV  DL,[BX]        ; DX=个位2
        INC  BX
        MOV  AL,[BX]        ; 十位5
        MUL  M2             ; AX=5×10=50
        ADD  DX,AX          ; DX=52
        INC  BX
        MOV  AL,[BX]        ; 百位6
        MUL  M1             ; AX=6×100=600
        ADD  DX,AX          ; DX=652
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

执行结果：DX=652=028CH。考点：按权展开（个位×1+十位×10+百位×100）；MUL 字节乘结果在 AX；累加用 DX 保存部分和。

题16（课件 p99-100，例4）

键盘输入两个一位十进制数，相加并把结果显示在屏幕上。

解答：

```
CODE    SEGMENT
        ASSUME CS:CODE
START:  MOV  AH,01H          ; 1号功能：键入第1个数
        INT  21H             ; AL=ASCII(回显)
        MOV  BL,AL           ; 暂存第1个数
        MOV  AH,01
        INT  21H             ; 键入第2个数
        ADD  AL,BL           ; 两ASCII直接相加
        MOV  AH,00H          ; AH清0供AAA进位
        AAA                  ; 非压缩加法调整→AH十位,AL个位
        OR   AX,3030H        ; 两位同时转ASCII
        MOV  BX,AX           ; BH=十位ASCII,BL=个位ASCII
        MOV  DL,BH
        MOV  AH,02
        INT  21H             ; 先显十位
        MOV  DL,BL
        MOV  AH,02H
        INT  21H             ; 再显个位
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

模拟输入 '5','7'：35H+37H=6CH；AAA：低半 C>9 → AL=6CH+6=72H、AH=1，AL&0F=2 → AX=0102H；OR 3030H → 3132H；显示 "12" ✓。考点：ASCII 相加后必须 AAA；OR AX,3030H 一条指令转两位。

题17（课件 p101-103，例5）

A 起存 10 个字节压缩 BCD 码，转换为 ASCII 码并显示在 CRT 上（每字节两位，字节间空格）。

解答：

```
DATA    SEGMENT
A       DB   34H,68H,76H,12H,90H,55H,23H,81H,47H,60H
N       EQU  $-A            ; =10
DATA    ENDS
STACK   SEGMENT PARA STACK 'STACK'
STA     DB   100 DUP(?)
STACK   ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA,SS:STACK
START:  MOV  AX,DATA
        MOV  DS,AX
        MOV  AX,STACK
        MOV  SS,AX
        MOV  BX,OFFSET A    ; 源指针
        MOV  CX,N           ; 循环10次
LP:     MOV  AL,[BX]        ; 取压缩BCD字节
        MOV  DH,AL          ; 备份(留低半用)
        PUSH CX             ; 保护循环计数(CX会被CL占用)
        MOV  CL,4
        SHR  AL,CL          ; 高半移到低4位
        POP  CX             ; 恢复计数
        OR   AL,30H         ; 高半转ASCII
        MOV  DL,AL
        MOV  AH,2
        INT  21H            ; 显示高位数字
        AND  DH,0FH         ; 取低半
        OR   DH,30H         ; 低半转ASCII
        MOV  DL,DH
        INT  21H            ; 显示低位数字(AH仍=2)
        MOV  DL,' '
        INT  21H            ; 显示空格
        INC  BX
        LOOP LP
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

首字节 34H → 显示 "3 4 "。考点：PUSH CX/POP CX 保护计数（SHR 需 CL）；高半 SHR 4、低半 AND 0FH；AH=2 可连续复用。

题18（课件 p104-106，例6）

将 AL 寄存器内容送屏幕显示：(1) 二进制形式；(2) 十六进制形式（分支版与循环版）。

解答：(1) 二进制（CX=8，ROL 把最高位旋入 CF）：

```
CODE    SEGMENT
        ASSUME CS:CODE
START:  MOV  CX,08H          ; 8位
NEXT:   ROL  AL,1            ; 最高位→CF，循环左移
        JC   NEXT1           ; CF=1显示'1'
        MOV  DL,30H          ; CF=0显示'0'
        JMP  NEXT2
NEXT1:  MOV  DL,31H
NEXT2:  MOV  AH,02H
        INT  21H             ; 显示一位
        LOOP NEXT            ; 8次后AL复原
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

(2) 十六进制分支版（高半、低半各一段）：

```
CODE    SEGMENT
        ASSUME CS:CODE
START:  MOV  BL,AL          ; 备份
        MOV  BH,AL
        MOV  CL,04H
        SHR  BL,CL          ; BL=高半字节
        CMP  BL,09H
        JA   NEXT1          ; >9转字母处理
        OR   BL,30H         ; 0~9→ASCII
        JMP  NEXT2
NEXT1:  ADD  BL,37H         ; A~F→ASCII('A'=41H=0AH+37H)
NEXT2:  MOV  DL,BL
        MOV  AH,02H
        INT  21H            ; 显示高位hex
        AND  BH,0FH         ; 取低半
        CMP  BH,09H
        JA   NEXT3
        OR   BH,30H
        JMP  NEXT4
NEXT3:  ADD  BH,37H
NEXT4:  MOV  DL,BH
        MOV  AH,02H
        INT  21H            ; 显示低位hex
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

(3) 十六进制循环版（CH=2 控制两遍，课件 p106）：

```
CODE    SEGMENT
        ASSUME CS:CODE
START:  MOV  BL,AL
        MOV  BH,AL
        MOV  CH,2           ; 两遍(高半、低半)
        MOV  CL,04H
        SHR  BL,CL          ; 先取高半
NEXT:   CMP  BL,09H
        JA   NEXT1
        OR   BL,30H
        JMP  NEXT2
NEXT1:  ADD  BL,37H
NEXT2:  MOV  DL,BL
        MOV  AH,02H
        INT  21H            ; 显示当前半字节
        AND  BH,0FH         ; 取低半
        MOV  BL,BH          ; 送入BL供下一遍
        DEC  CH
        JNZ  NEXT           ; 第二遍
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

设 AL=0B7H：(1) 显示 10110111；(2)(3) 显示 B7。考点：ROL+JC 逐位取二进制；hex 转换 0~9 加 30H、A~F 加 37H；循环版 CH 计数与 CL 移位分工。

题19（课件 p107-108，例7）

把 M1 起 100 个单元的数字 ASCII 码转换成组合 BCD 码，仍放在 M1 起的 50 个单元中（原地压缩）。

解答：

```
DATA    SEGMENT
M1      DB   100 DUP(?)     ; 100个数字ASCII
DATA    ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA
START:  MOV  AX,DATA
        MOV  DS,AX
        MOV  SI,OFFSET M1   ; 读指针(步长2)
        MOV  DI,OFFSET M1   ; 写指针(步长1)
        MOV  CH,32H         ; 50组
        MOV  CL,04H         ; 移位次数
TTT:    MOV  AL,[SI]        ; 第1个ASCII
        AND  AL,0FH         ; →低半BCD
        MOV  BL,[SI+1]      ; 第2个ASCII
        SHL  BL,CL          ; →高半BCD
        OR   AL,BL          ; 合并为压缩BCD
        MOV  [DI],AL        ; 存1字节
        INC  SI
        INC  SI             ; 读指针+2
        INC  DI             ; 写指针+1
        DEC  CH
        JNZ  TTT            ; 50组
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

模拟：M1='1','2'（31H,32H）→ AL=01H、BL=20H → OR=21H 存 M1[0]。原地安全性：第 i 组读 M1+2i、写 M1+i，读指针始终超前写指针，不会覆盖未读数据。考点：CH 计数/CL 移位分工；原地压缩的指针步长关系。

（二）课堂习题（课件 p109）

题20（课件 p109 习题1，答案 p110-111）

从 ARRAY 开始的单元中存有 20 个 16 位无符号数，编一完整程序找出其中最大数并存入 MAX 单元。

解答： 课件答案：

```
DATA    SEGMENT
ARRAY   DW   10,20,30,12,5,7,90,44,66,88,1,2,3,4,6,8,9,11,13,15  ; 20个
MAX     DW   ?
DATA    ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA
START:  MOV  AX,DATA
        MOV  DS,AX
        MOV  CX,20          ; 存疑：应为19，见下
        LEA  SI,ARRAY
        MOV  AX,[SI]        ; 当前最大=第1个数
L1:     CMP  AX,[SI+2]      ; 与下一个字比较
        JA   NEXT           ; 无符号：AX大则跳过
        MOV  AX,[SI+2]      ; 否则更新
NEXT:   INC  SI
        INC  SI             ; 字指针+2
        LOOP L1
        MOV  MAX,AX
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

存疑：20 个数只需 19 次比较，课件 CX=20 使最后一趟 SI 指向 ARRAY+38（第20个数），`[SI+2]` 越界读到 MAX 单元（DW ? 初值不定）。若 MAX 初值大于真实最大数则结果错误。修正：`MOV CX,19`。其余逻辑正确：无符号用 JA、字步长 2。示例数据最大数 90 → MAX=90=005AH。

题21（课件 p109 习题2，答案 p112）

将 AX 寄存器的内容送屏幕显示输出（以十六进制形式）。

解答：

```
CODE    SEGMENT
        ASSUME CS:CODE
START:  MOV  BX,AX          ; 待显示字
        MOV  CH,4           ; 4个hex位
        MOV  CL,4           ; 每次循环左移4位
NEXT:   ROL  BX,CL          ; 最高nibble旋到最低nibble
        MOV  DL,BL
        AND  DL,0FH         ; 取当前最低nibble
        CMP  DL,09H
        JA   AD1            ; >9转字母
        OR   DL,30H         ; 0~9→ASCII
        JMP  AD2
AD1:    ADD  DL,37H         ; A~F→ASCII
AD2:    MOV  AH,02H
        INT  21H            ; 显示一位
        DEC  CH
        JNZ  NEXT           ; 4位
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

执行分析：ROL BX,4 每次把下一个高位 nibble 转到 BL 低 4 位，故从最高位到最低位依次显示。设 AX=1A3FH → 显示 "1A3F"。与题18(3) 对比：本题用 ROL 旋转取位，题18 用 SHR+AND 拆半，二者等价常考。

题22（课件 p109 习题3，答案 p113-114）

DATA1 起连续存放 50 个组合 BCD 码，转换成 ASCII 码存 DATA2 起存储区；高位 BCD 转成的 ASCII 放地址较高的单元。

解答：

```
DATA    SEGMENT
DATA1   DB   50H,40H,25H,11H,22H,33H,44H,55H,66H,77H,88H,99H,01H,02H,03H,04H,05H,06H,07H,08H,09H,1
DATA2   DB   100 DUP(?)      ; 100个ASCII
DATA    ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA
START:  MOV  AX,DATA
        MOV  DS,AX
        LEA  SI,DATA1        ; 源指针
        LEA  DI,DATA2        ; 目的指针
        MOV  CH,50           ; 50字节
        MOV  CL,4            ; 移位次数
L1:     MOV  AL,[SI]         ; 取压缩BCD
        MOV  BL,AL           ; 备份
        AND  AL,0FH          ; 低半
        OR   AL,30H          ; →ASCII
        MOV  [DI],AL         ; 低半ASCII存低地址
        INC  DI
        SHR  BL,CL           ; 高半移到低4位
        OR   BL,30H          ; →ASCII
        MOV  [DI],BL         ; 高半ASCII存高地址
        INC  SI
        INC  DI
        DEC  CH
        JNZ  L1
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

首字节 50H → DATA2[0]=30H('0')、DATA2[1]=35H('5')，满足"高位 BCD 的 ASCII 在较高地址"。考点：拆半顺序与存放地址对应；CH 计数/CL 移位。

题23（课件 p109 习题4，答案 p115-116）

DATA1 起连续存放 50 个组合 BCD 码，转换成 ASCII 码仍存 DATA1 起存储区（原地）；高位 BCD 的 ASCII 放地址较高的单元。

解答：课件答案（p115-116，双栏已按图还原）：

```
DATA    SEGMENT
DATA1   DB   10H,20H,12H,34H,56H,78H,90H,11H,22H,33H,44H,55H,66H,77H,88H,99H,01H,02H,03H,04H,05H,0
        DB   50 DUP(?)      ; 原地扩展预留50字节
DATA    ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA
START:  MOV  AX,DATA
        MOV  DS,AX
        MOV  CH,50
        MOV  CL,4
        LEA  SI,DATA1+49    ; 读指针从最后一个BCD开始
        LEA  DI,DATA1+99    ; 写指针从最后地址开始
L1:     MOV  AL,[SI]
        MOV  BL,AL
        SHR  AL,CL          ; 高半
        OR   AL,30H
        MOV  [DI],AL        ; 高半ASCII存高地址
        DEC  DI
        AND  BL,0FH         ; 低半
        OR   BL,30H
        ; 存疑：课件此处漏 MOV [DI],BL，低半ASCII未写入！
        DEC  DI
        DEC  SI
        DEC  CH
        JNZ  L1
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

存疑与修正：课件 p116 图中 `OR BL,30H` 之后直接 `DEC DI`，缺少 `MOV [DI],BL`，导致低半 ASCII 丢失。修正版在 `OR BL,30H` 后补 `MOV [DI],BL`。原地安全性：从后往前处理，第 i 轮（i=0~49）读 DATA1+49-i，写 DATA1+99-2i（高半）与 DATA1+98-2i（低半）；因 98-2i ≥ 49-i 恒成立，写地址不会落到读地址之前；末轮 i=49 读 +0、写 +1 与 +0，其中 +0 字节已在本轮开头读入 AL/BL，故不会被破坏。修正后原 DATA1[0]=10H 最终变为 DATA1[0]=30H('0')、DATA1[1]=31H('1')。

（三）教材习题（课件附答案页，题干见教材）

题24（课件 p126，5-1）

（题干见教材：给出数据段中 VR1~VR5 与 COUNT 的定义，求各变量占用的存储字节数。）课件答案：VR1=2，VR2=8，COUNT 不分配存储空间，VR3=40，VR4=22，VR5=12。

解答： 按数据定义宽度规则解读答案： - VR1=2 → 字变量（DW 一项）； - VR2=8 → 双字两项（DD 2 项）或四字一项（DQ）； - COUNT 不分配存储空间 → 由 EQU（或 =）定义的符号常量，只作汇编期替换，不占内存； - VR3=40 → 如 DB 40 DUP(?) 或 DW 20 DUP(?)； - VR4=22 → 如 DB 22 DUP(?) 或 DW 11 DUP(?) 或混合定义合计 22 字节； - VR5=12 → 如 DD 3 DUP(?) 或 DW 6 DUP(?)。 方法要点：字节数 = 各数据项宽度之和，DUP 逐层相乘；EQU/= 不分配单元。本题考点即"哪些伪指令分配存储、哪些不分配"。

题25（课件 p126，5-2）

（题干见教材：对给定变量/标号求分析运算符的值。）课件答案：(1)AX=1 (2)BX=2 (3)CX=100 (4)SI=200 (5)DI=1。

解答： 答案对应的典型设问与推理： - (1) AX=1：`MOV AX,TYPE 字节变量` → BYTE 类型值 1； - (2) BX=2：`MOV BX,TYPE 字变量` → WORD 类型值 2； - (3) CX=100：`MOV CX,LENGTH 变量`（该变量为 100 DUP 定义）→ 最外层 DUP 的 n=100； - (4) SI=200：`MOV SI,SIZE 变量`（100 DUP 的字变量）→ LENGTH×TYPE=100×2=200； - (5) DI=1：`MOV DI,LENGTH 变量`（非 DUP 定义，如 DW 1,2,3）→ 返回 1。 考点：TYPE/LENGTH/SIZE 三运算符的返回值规则（课件 p15-17），尤其 LENGTH 对非 DUP 定义返回 1、SIZE=LENGTH×TYPE。

题26（课件 p126，5-3）

（题干见教材：给出数据定义，画出内存分配图。）课件答案图见笔记 §15 例4（slide_126）。按图还原内存布局（两栏连续表示同一数据区，左栏 11 字节接右栏，共 20 字节）：

偏移	内容	归属
0~3	01H,02H,03H,04H	A（DB 1,2,3,4）
4~7	41H,42H,43H,44H	B（DB 'ABCD'）
8~11	00H,00H,00H,00H	C（4 个 0 字节）
12~15	21H,00H,02H,03H	X（DW 21H,302H，低字节低地址）
16~19	0CDH,0ABH,00H,00H	Y（DD 0000ABCDH）

解答： 读图方法：① 字符串按 ASCII 顺排（'A'=41H…）；② 字/双字低字节在低地址（X 的字 0021H 存为 21H,00H；Y 的双字 0000ABCDH 存为 CDH,ABH,00H,00H）；③ 变量名标在其首字节处。存疑：课件图以两栏换行表示连续内存，C 与 X 的边界按图读为 C 占 4 个 0 字节；若教材原题为 3 DUP(0)，则右栏首 0 应归 X，X 的拆字方式随之变化，以教材原题定义为准。

题27（课件 p127，5-4）

（题干见教材：按要求写出数据段定义。）课件答案：

```
DATA    SEGMENT
ARRAY   DB   10 DUP(29H)     ; 10字节，初值29H
ALPHA   DW   -28,6,45,39,3   ; 5个字=10字节
BUFFER  DB   100 DUP(?)      ; 100字节预留
DATA    ENDS
```

解答（补充内存布局计算）： - ARRAY：偏移 0~9，10 字节，每字节 29H； - ALPHA：偏移 10~19，10 字节；各字的机器存储（低字节低地址）：-28=FFE4H→E4H,FFH；6→06H,00H；45=002DH→2DH,00H；39=0027H→27H,00H；3→03H,00H； - BUFFER：偏移 20~119，100 字节不初始化。 考点：负数以补码存放（-28=FFE4H）；DUP 与初值列表混合定义的字节累计。

题28（课件 p127，5-7）

（题干见教材：程序阅读题，问屏幕显示内容。）课件答案：屏幕显示 **123ABC**。

解答： 结果分析：程序依次输出字符 '1','2','3','A','B','C'。典型实现（还原示例，非课件原文）：数据段 `MSG DB '123ABC$'`，代码 `MOV DX,OFFSET MSG / MOV AH,9 / INT 21H`；或六次 2 号功能调用依次送 DL=31H,32H,33H,41H,42H,43H。评分要点：答出显示内容为 123ABC（无空格无换行），并说明 9 号功能遇 '$' 停止或 2 号功能逐字符输出。

题29（课件 p127，5-11）

（题干见教材：程序阅读题，(1) 问程序功能；(2) 问结果。）课件答案：(1) 20 之前的偶数相加；(2) 110 或 6EH。

解答： (1) 功能：累加偶数 2+4+6+…+20。 (2) 结果：2+4+…+20 = (2+20)×10/2 = **110 = 6EH**。注意按答案 110 可知累加**含 20**（若不含 20 则为 90），课件"20之前"应理解为"到20为止"。 还原示例程序（非课件原文）：

```
        MOV  CX,10          ; 10个偶数
        MOV  AX,0           ; 和清0
        MOV  BX,2           ; 首个偶数
LP:     ADD  AX,BX          ; 累加
        ADD  BX,2           ; 下一偶数
        LOOP LP             ; 结果AX=110=006EH
```

考点：循环累加三要素（和初值、步长、次数）；十进制与十六进制结果互换（110=6EH）。

题30（课件 p128-129，5-12）

（题干见教材：ARRAY 起 10 个字数据，求最小数存 MIN 单元。）课件答案代码：

```
DATA    SEGMENT
ARRAY   DW   1234H,5678H,0A34H,0,1,2,3,4,5,6   ; 10个(课件略写)
MIN     DW   ?
DATA    ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA
START:  MOV  AX,DATA
        MOV  DS,AX
        MOV  SI,OFFSET ARRAY
        MOV  DI,OFFSET MIN
        MOV  CX,09          ; 9次比较
        MOV  AX,WORD PTR [SI]   ; 当前最小=第1个字
LP1:    CMP  AX,WORD PTR [SI+1] ; 存疑：字数组应为[SI+2]
        JB   LP2            ; 无符号：AX小则跳过
        MOV  AX,WORD PTR [SI+1] ; 存疑：应为[SI+2]
LP2:    INC  SI             ; 存疑：字指针应+2
        DEC  CX
        JNZ  LP1
        MOV  WORD PTR [DI],AX   ; 存最小数
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

存疑与修正：ARRAY 为 DW 字数组，课件用 `[SI+1]` 取"下一个字"且 `INC SI` 步长 1，会读到跨字的错位组合（如 1234H,5678H 存储为 34 12 78 56，[SI+1] 读出 7812H），逻辑错误。修正版：

```
        MOV  CX,9
        MOV  AX,[SI]
LP1:    CMP  AX,[SI+2]      ; 下一个字
        JB   LP2
        MOV  AX,[SI+2]
LP2:    ADD  SI,2           ; 字步长2
        DEC  CX
        JNZ  LP1
        MOV  [DI],AX
```

修正后示例数据最小数 0 → MIN=0。考点：字数组指针步长 2；n 个数比 n-1 次；无符号最小用 JB。

题31（课件 p130-131，5-13）

（题干见教材：DAT 起 100 个字节数据，统计其中奇数个数存 RESULT 单元。）课件答案：

```
DATA    SEGMENT
DAT     DB   34H,68H,0F5H,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,
RESULT  DB   0              ; 奇数计数初值0
DATA    ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA
START:  MOV  AX,DATA
        MOV  DS,AX
        MOV  SI,OFFSET DAT
        MOV  CX,100         ; 100个数据
LP:     MOV  AL,[SI]
        TEST AL,01          ; 测最低位(课件写01,即1)
        JZ   NEXT           ; 最低位0→偶数跳过
        INC  RESULT         ; 奇数计数+1
NEXT:   INC  SI
        DEC  CX
        JNZ  LP
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

执行分析：TEST AL,1 只置标志不改变 AL；ZF=1 表示最低位 0（偶数）。示例数据中奇数为 F5H 及 1,3,…,97 共 50 个 → RESULT=50=32H。考点：TEST 判奇偶；内存计数单元直接 INC；CX=100 全量扫描（统计类不需要 n-1）。

题32（课件 p132，5-14）

（题干见教材：从键盘输入字符，将小写字母转成大写显示，直到输入 '$' 结束。）课件答案：

```
CODE    SEGMENT
        ASSUME CS:CODE
START:  MOV  AH,01          ; 1号功能键入一字符(回显)
        INT  21H
        CMP  AL,24H         ; 是'$'吗
        JZ   STOP           ; 是→结束
        SUB  AL,20H         ; 小写→大写(ASCII差20H)
        MOV  DL,AL
        MOV  AH,02
        INT  21H            ; 显示大写字符
        JMP  START          ; 继续读下一个
STOP:   MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

执行分析：输入 'a'(61H) → 减 20H → 41H 显示 'A'；输入 '$'(24H) → JZ 退出。注意 1 号功能本身回显原字符，故屏幕先现小写再现大写。考点：大小写差 20H；循环读字符的出口条件；JMP START 构成条件控制循环。

（四）课件附带的第4章习题答案（p117-125，题干见教材第4章）

题33（课件 p117，4-1）

（题干见教材：段地址/物理地址相关填空。）课件答案：(1) 3000H，DS 的值；(2) 4000H；(3) 6020H；(4) 4000H；(5) 6050H。

解答： 按物理地址 = 段基址×16 + 偏移地址 复核方法：如段基址 3000H、偏移 20H → 物理 30020H；答案 (3)6020H、(5)6050H 形如"段基址+偏移"的组合结果。本题属第4章寻址计算，课件仅附答案，复习时以第4章笔记的段址换算公式为准。

题34（课件 p117，4-2）

（题干见教材：判断指令书写正误。）课件答案：**全错**。

解答： 课件判定各小题指令全部错误。第4章常见非法写法供对照复习：立即数送段寄存器（MOV DS,1234H）、两操作数同为内存（MOV [AX],[BX]）、操作数宽度不匹配（MOV AX,BL）、CS 作目的操作数（MOV CS,AX）、立即数作目的操作数等。评分要点：答"全错"并能举出对应非法类型。

题35（课件 p117，4-3）

（题干见教材：PUSH/POP 序列后寄存器与 SP 的值。）课件答案：(1) AX=3355H，SP=1FFEH；(2) AX=3355H，CX=3355H，DX=4466H，SP=2000H。

解答： 推理：SP 初值 2000H；(1) 执行一次 PUSH 后 SP=2000H-2=1FFEH，被压字为 3355H（随后 POP 回 AX 得 3355H）；(2) 再经 PUSH/POP 配对后 SP 恢复 2000H，AX=CX=3355H、DX=4466H 说明栈中两字分别弹入 CX、DX。考点：PUSH 先 SP-2 再写栈、POP 先读栈再 SP+2；压弹配对 SP 复原。

题36（课件 p118，4-4）

（题干见教材：按要求写指令序列。）课件答案： (1) MOV AX,1234H / MOV DS,AX；(2) ADD AX,5678H；(3) MOV AX,DATAX / ADD DATAY,AX；(4) AND AX,0FFFH；(5) OR BX,0003H；(6) XOR CX,00F8（课件原文如此，存疑：应为 00F8H）。

解答： 逐条说明：(1) 段寄存器不能收立即数，须借 AX 中转；(2) 累加立即数；(3) 内存加内存须借寄存器中转；(4) AND 清高 4 位（屏蔽）；(5) OR 置位（bit0、bit1 置 1）；(6) XOR 指定位取反。考点：段寄存器装填、内存-内存运算中转、AND/OR/XOR 的屏蔽/置位/取反用途。

题37（课件 p118，4-5）

（题干见教材：BCD 调整指令执行后的标志。）课件答案：(1) AF=1，CF=1；(3) AF=1，CF=1。

解答： DAA/AAA 类调整在产生十进制进位时置 CF=1、半进位 AF=1。如压缩 BCD 99H+01H=9AH → DAA → 00H 且 CF=1、AF=1。评分要点：答出调整指令进位标志含义（CF=十进制进位，AF=低半进位）。

题38（课件 p119，4-6）

（题干见教材：BUF 起存两个压缩 BCD 字，求差存后两字节。）课件给出程序与数据：BUF 起字节为 43H,65H,72H,46H,?,?（即字 6543H 与 4672H）。

```
        LEA   SI,BUF          ; SI=BUF首址
        MOV   AX,[SI]         ; AX=6543H(被减数)
        MOV   BX,[SI+2]       ; BX=4672H(减数)
        SUB   AL,BL           ; 低字节43H-72H=D1H,CF=1
        DAS                   ; 压缩BCD减调整→AL=71H,CF=1
        MOV   [SI+4],AL       ; 存差低字节71H
        MOV   AL,AH           ; AL=65H
        SBB   AL,BH           ; 65H-46H-1=1EH,AF=1
        DAS                   ; 调整→AL=18H
        MOV   [SI+5],AL       ; 存差高字节18H
        HLT
```

解答（完整跟踪）： - 43H-72H：低半 3-2=1 无半借位，高半 4-7 借位 → D1H、CF=1；DAS 高半>9 减 60H → 71H（即 100-29=71，借位 1）✓ - 65H-46H-1：低半 5-6-1 借位 → 1EH、AF=1；DAS 低半 E>9 减 6 → 18H ✓ - 结果 [SI+4]=71H、[SI+5]=18H，即 BCD 差 1871；验算 6543-4672=1871 ✓，最终 CF=0。 考点：DAS 紧跟 SUB/SBB；多字节 BCD 减法用 SBB 串借位。

题39（课件 p120，4-7）

（题干见教材：BCD 运算各小题结果。）课件答案：(2) 0011 0110 1010 0101 与 0000 0110 1101 0100，标注 06D4H、CF=1；(4) 364AH，CF=1；(6) 362DH，CF=1；(8) 06A5H，CF=1；(10) 66D4H，CF=1。

解答： 课件仅附各小题结果（题干见教材）：(2) 给出两操作数的二进制形式 0011 0110 1010 0101（36A5H）与 0000 0110 1101 0100（06D4H），并标注 06D4H、CF=1；其余各小题结果如上。复习要点：这些是 BCD 运算（经 DAA/DAS 调整）的结果与标志，CF=1 表示产生十进制进位/借位，而非二进制进位；答题以课件数值为准。

题40（课件 p121-122，4-8）

（题干见教材：把 AL 的 8 位二进制位反序送入 BL。）课件程序：

```
        MOV  CL,08          ; 8位
        MOV  AL,01110001B   ; 71H
        MOV  BL,00000000B   ; 结果清0
DDD:    SHL  AL,1           ; AL最高位→CF
        RCR  BL,1           ; CF→BL最高位，BL右移
        DEC  CL
        JNZ  DDD            ; 8次
        HLT
```

解答（跟踪）：每轮 SHL 把 AL 的当前最高位移入 CF，RCR 把 CF 移入 BL 最高位并使 BL 右移，8 轮后 AL 的位序完全反转进 BL。AL=01110001B → BL=**10001110B=8EH**，AL=00H。课件 p121 给出中间轨迹：第1轮 SHL 后 AL=11100010、CF=0，RCR 后 BL=00000000；第2轮 SHL 后 AL=11000100、CF=1，RCR 后 BL=10000000……与程序一致。考点：SHL+RCR 组合实现位反序（CF 作桥梁）。

题41（课件 p123，4-9）

（题干见教材：指令序列执行后寄存器值。）课件答案：AX=7D00H，BX=7D01H，CL=03；另有 0A0AH-00FFH 的运算。

解答： 课件仅附结果。可复核部分：0A0AH-00FFH=090BH（若为 BCD 调整题则另论）。AX=7D00H、BX=7D01H 形如地址/指针计算结果，CL=03 为移位次数。题干见教材第4章，复习以课件答案数值为准。

题42（课件 p123，4-10）

（题干见教材：串操作填空。）课件答案：MOV DI,1000H / REP MOVSB / REPNZ SCASB / DEC DI。

解答： 逐句说明：`MOV DI,1000H` 设目的串指针；`REP MOVSB` 按 CX 次数重复字节传送（DS:SI→ES:DI，SI/DI 自动±1）；`REPNZ SCASB` 重复扫描直到 AL 与 [DI] 相等或 CX=0（ZF=1 停）；`DEC DI` 把 DI 回退到匹配字符位置（SCASB 停止时 DI 已指向下一字节）。考点：REP/REPNZ 前缀与 MOVSB/SCASB 的配合、DF 方向、停止后指针回退。

题43（课件 p124，4-11）

（题干见教材：统计 AX 中 1 的个数。）课件程序：

```
        MOV  CL,16          ; 16位
        MOV  BL,0           ; 计数清0
BB:     ROL  AX,1           ; 最高位→CF
        JNC  NEXT           ; CF=0跳过
        INC  BL             ; CF=1计数+1
NEXT:   DEC  CL
        JNZ  BB             ; 16次
        MOV  CL,BL          ; 个数送CL
        HLT
```

解答：ROL 循环左移 16 次遍历 AX 每一位，CF=1 时 BL 加 1；结束后 AX 复原、CL=AX 中 1 的个数。如 AX=0003H → CL=2。考点：ROL+JNC 逐位统计；16 次后寄存器复原。

题44（课件 p124，4-12）

（题干见教材：转移指令目标标号填空。）课件答案：L1、L1、L2、L5、L5。

解答：课件仅附五个空的答案标号。复习要点：条件转移与 LOOP 的目标为短标号（-128~127），JMP 可段内直接/间接；按程序 flow 逐条核对转移条件即可得目标。题干见教材。

题45（课件 p125，4-13）

（题干见教材：程序阅读，说明功能。）课件程序：

```
        MOV  SI,OFFSET BUF  ; 取数据地址
        MOV  AL,[SI]        ; AL=字节数据
        CMP  AL,20          ; 与20比较
        JB   NEXT           ; 无符号<20转NEXT
        SUB  AL,20          ; >=20：减20
        JMP  STOP
NEXT:   MOV  BL,3
        MUL  BL             ; <20：乘3(AX=AL*3)
STOP:   HLT
```

解答：功能：分段函数。设 BUF 单元字节为 x（无符号）：x≥20 时 AL=x-20；x<20 时 AX=3x。如 x=25 → AL=5；x=7 → AX=21。考点：CMP+JB 实现分支；MUL BL 字节乘结果在 AX；JMP STOP 保证两分支互斥。

二、考点补充题

题46【补充•考点：循环与数组求和】

ARRAY 起存 10 个 16 位无符号数，求和（和可能超过 16 位），结果以双字存 SUM 单元（低字在低地址）。写完整程序。

解答：

```
DATA    SEGMENT
ARRAY   DW   1000,2000,3000,4000,5000,6000,7000,8000,9000,60000  ; 和=105000>65535
SUM     DD   ?              ; 双字结果
DATA    ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA
START:  MOV  AX,DATA
        MOV  DS,AX
        LEA  SI,ARRAY       ; 源指针
        MOV  CX,10          ; 10个字
        XOR  AX,AX          ; 和低字清0
        XOR  DX,DX          ; 和高字清0
LP:     ADD  AX,[SI]        ; 累加低字
        ADC  DX,0           ; 进位入高字
        INC  SI
        INC  SI             ; 字步长2
        LOOP LP
        MOV  WORD PTR SUM,AX    ; 存低字
        MOV  WORD PTR SUM+2,DX  ; 存高字
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

验证：和=105000=00019A28H → SUM 低字 9A28H、高字 0001H。考点：ADD+ADC 0 构成 32 位累加；字步长 2；DD 结果分两次字存储。

题47【补充•考点：数据定义与分析运算符】

设有数据段：

```
DATA    SEGMENT
BUF1    DB   20 DUP(?)
BUF2    DW   5 DUP(1,2)
BUF3    DD   3 DUP(?)
STR     DB   'HELLO$'
DATA    ENDS
```

求：(1) 各变量的 TYPE、LENGTH、SIZE；(2) 各变量的偏移地址；(3) 数据段总字节数。

解答： (1) 运算符值表：

变量	TYPE	LENGTH	SIZE
BUF1	1	20	20
BUF2	2	5	10
BUF3	4	3	12
STR	1	1	1

注意 BUF2 每项 DUP 内含 2 个字，实际占 5×2×2=20 字节，但 LENGTH 只返回最外层 DUP 的 5、SIZE=5×2=10（课件 p16-17 规则）；STR 非 DUP 定义 LENGTH=1、SIZE=1（实际 6 字节）。 (2) 偏移：BUF1=0；BUF2=20；BUF3=20+20=40；STR=40+12=52。 (3) 总字节=20+20+12+6=58。 考点：LENGTH/SIZE 的"最外层 DUP"陷阱与实际占用字节数的区别；偏移按实际字节累计。

题48【补充•考点：字符串处理（原地逆序）】

STR 起存放一串字符（长度 LEN 由 EQU $-STR 给出），编程将其原地逆序。

解答:

```
DATA    SEGMENT
STR     DB   'ABCDEF'        ; 示例6字符
LEN     EQU  $-STR           ; =6
DATA    ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA
START:  MOV  AX,DATA
        MOV  DS,AX
        LEA  SI,STR          ; 头指针
        LEA  DI,STR+LEN-1    ; 尾指针
        MOV  CX,LEN/2        ; 交换次数=6/2=3
LP:     MOV  AL,[SI]         ; 取头字符
        XCHG AL,[DI]         ; 与尾字符交换(AL得尾字符)
        MOV  [SI],AL         ; 写回头位置
        INC  SI              ; 头指针后移
        DEC  DI              ; 尾指针前移
        LOOP LP
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

跟踪：第1轮 A↔F → FBCDEA；第2轮 B↔E → FEDCBA? 逐步：F B C D E A → 交换 B,E → F E C D B A；第3轮 C↔D → F E D C B A。结果 'FEDCBA' ✓。考点：首尾双指针对撞；XCHG 一条指令完成读尾+暂存；次数=n/2（奇数长度中间字符不动）。

题49【补充•考点：分支循环综合（三分类统计）】

BUF 起存 10 个 8 位带符号数，统计正数、零、负数的个数，分别存 POS、ZER、NEG 单元。

解答:

```
DATA    SEGMENT
BUF     DB   5,-3,0,12,-7,0,8,-1,4,6
N       EQU  $-BUF          ; =10
POS     DB   0              ; 正数计数
ZER     DB   0              ; 零计数
NEG     DB   0              ; 负数计数
DATA    ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA
START:  MOV  AX,DATA
        MOV  DS,AX
        LEA  SI,BUF
        MOV  CX,N           ; 10个
LP:     MOV  AL,[SI]
        CMP  AL,0           ; 与0比较(带符号)
        JG   ISPOS          ; >0正数
        JL   ISNEG          ; <0负数
        INC  ZER            ; 否则零
        JMP  NEXT
ISPOS:  INC  POS
        JMP  NEXT
ISNEG:  INC  NEG
NEXT:   INC  SI
        LOOP LP
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

验证：正数 5,12,8,4,6 → POS=5；负数 -3,-7,-1 → NEG=3；零 2 个 → ZER=2；5+3+2=10 ✓。考点：CMP 0 后 JG/JL 双分支+ fall-through 零分支；三分支互斥用 JMP NEXT 收口；带符号语义。

题50【补充•考点：宏定义与宏展开】

定义一个带两个参数的宏 ADDM，功能为把两个字节变量相加结果留 AL；写出宏定义、调用语句及展开后的指令序列，并说明宏与子程序的区别。

解答：

```
ADDM    MACRO X,Y           ; 形参X,Y
        MOV  AL,X           ; 宏体
        ADD  AL,Y
        ENDM
; 数据段：A DB 12H  B DB 34H
; 调用：
        ADDM A,B            ; 实参代入
; 汇编时展开为：
        MOV  AL,A           ; AL=12H
        ADD  AL,B           ; AL=46H
```

执行结果 AL=46H。区别（评分要点）：① 宏在汇编期展开，每次调用插入一份宏体，无 CALL/RET 开销但代码量随调用次数增长；子程序运行期调用，代码仅一份，有压栈返回开销；② 宏参数是文本替换，可传寄存器/内存/立即数，子程序参数靠寄存器或内存约定传递；③ 短小频繁代码宜用宏，较长功能宜用子程序（课件 p44-46）。

题51【补充•考点：DOS功能调用综合】

编程：用 0AH 号功能从键盘输入一字符串（≤79 字符），回车后先换行，再原样显示该字符串，最后显示其实际长度（设<10）。

解答：

```
DATA    SEGMENT
BUF     DB   80              ; 容量80
        DB   ?               ; 系统回填实际长度
        DB   80 DUP(?)       ; 内容区(0DH结束)
CRLF    DB   0DH,0AH,'$'     ; 换行串
DATA    ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA
START:  MOV  AX,DATA
        MOV  DS,AX
        MOV  DX,OFFSET BUF
        MOV  AH,0AH
        INT  21H             ; 键入字符串
        MOV  BL,BUF+1        ; 取实际长度
        MOV  BH,0            ; BX=长度
        MOV  SI,OFFSET BUF+2
        ADD  SI,BX           ; SI指向0DH(串尾下一字节)
        MOV  BYTE PTR [SI],'$'  ; 回车符替换为'$'供9号使用
        MOV  DX,OFFSET CRLF
        MOV  AH,9
        INT  21H             ; 换行
        MOV  DX,OFFSET BUF+2
        MOV  AH,9
        INT  21H             ; 显示输入内容
        MOV  DL,BUF+1        ; 长度数字
        OR   DL,30H          ; 转ASCII(<10)
        MOV  AH,2
        INT  21H             ; 显示长度
        MOV  AH,4CH
        INT  21H
CODE    ENDS
        END  START
```

模拟输入 "ABC" 回车：BUF+1=3，BUF+2..4='ABC'，BUF+5=0DH 被改为 '$'；换行后显示 ABC，再显示 '3'。考点：0AH 与 9 号结束符差异（0DH→'$' 替换技巧）；实际长度在 BUF+1；长度转 ASCII。

题52【补充•考点：子程序与现场保护（拔高）】

编写子程序 HEXOUT：入口 AL=一个字节数，功能以两位十六进制字符显示其值；要求用寄存器传参、保护所用寄存器、正确返回。并给出调用示例。

解答：

```
; 调用示例（主程序内）：
        MOV  AL,0B7H
        CALL HEXOUT          ; 屏幕显示 B7
; 子程序：
HEXOUT  PROC NEAR
        PUSH BX              ; 现场保护
        PUSH CX
        PUSH DX
        MOV  BL,AL           ; 备份原值
        MOV  BH,AL
        MOV  CH,2            ; 两遍(高半/低半)
        MOV  CL,4
        SHR  BL,CL           ; 先取高半字节
HN:     CMP  BL,09H
        JA   HL              ; >9转字母
        OR   BL,30H          ; 0~9→ASCII
        JMP  HP
HL:     ADD  BL,37H          ; A~F→ASCII
HP:     MOV  DL,BL
        MOV  AH,02H
        INT  21H             ; 显示一位
        AND  BH,0FH          ; 取低半
        MOV  BL,BH
        DEC  CH
        JNZ  HN              ; 第二遍
        POP  DX              ; 恢复现场(与PUSH反序)
        POP  CX
        POP  BX
        RET                  ; 返回主程序
HEXOUT  ENDP
```

执行分析：AL=B7H → 第1遍 BL=0BH→'B' 显示；第2遍 BL=07H→'7' 显示；POP 顺序与 PUSH 相反保证 BX/CX/DX 复原；RET 弹出 CALL 压入的 IP。考点：寄存器传参（AL 入口、无出口）；PUSH/POP 成对反序；子程序内循环 CH/CL 分工；NEAR 过程与主程序同段。若主程序为 FAR 框架（课件 p51），本子程序仍可为 NEAR，因 CALL 为段内调用。