

3.3 试编写 10 个字(16 位二进制数)之和的程序。

```

1  data segment
2  myadd dw 0001h,0002h,0003h,0004h,0005h,0006h,0007h,0008h,0009h,000Ah
3  cont dw 9
4  data ends
5  code segment
6  assume cs:code,ds:data
7  process:
8  mov ax,data
9  mov ds,ax
10 clc
11 cld
12 mov si,offset myadd
13 mov di,offset myadd+2
14 mov cx,cont
15 mov ax,[si]
16 process1: adc ax,[di]
17 inc di
18 inc di
19 loop process1
20 mov ah,4ch
21 code ends
22 end process

```

DOSBox 0.74-3, Cpu speed: 3000 cycles, F...

```

AX=0037 BX=0000 CX=0001 DX=0000 SP=0000 BP=0000 SI=0000 DI=0012
DS=0788 ES=0778 SS=0787 CS=078A IP=0015  NU UP EI PL NZ AC PO NC
078A:0015 47          INC     DI
-T

AX=0037 BX=0000 CX=0001 DX=0000 SP=0000 BP=0000 SI=0000 DI=0013
DS=0788 ES=0778 SS=0787 CS=078A IP=0016  NU UP EI PL NZ NA PO NC
078A:0016 47          INC     DI
-T

AX=0037 BX=0000 CX=0001 DX=0000 SP=0000 BP=0000 SI=0000 DI=0014
DS=0788 ES=0778 SS=0787 CS=078A IP=0017  NU UP EI PL NZ NA PE NC
078A:0017 E2Fa      LOOP    0013
-T

```

$12 + \dots + 10 = 45 = (37)_{16}$

```

AX=0037 BX=0000 CX=0000 DX=0000 SP=0000 BP=0000 SI=0000 DI=0014
DS=0788 ES=0778 SS=0787 CS=078A IP=0019  NU UP EI PL NZ NA PE NC
078A:0019 B44C      MOV     AH,4C
-T

```

3.2 若 1KB 的数据存放在 TABLE 为首地址的主存区域，试编程序将该数据块搬到以 NEXT 为首地址的主存区域中。

```

1  DATA SEGMENT ;数据段
2  TABLE DB 256 DUP('FFFF');建立1024B的数据
3  NEXT DB 1024 DUP(?);建立1024B的数据空间
4  DATA ENDS
5
6  CODE SEGMENT ;程序段
7  ASSUME CS:CODE,DS:DATA
8  START:
9  MOV AX,DATA
10 MOV DS,AX
11 CLD
12 MOV SI,OFFSET TABLE
13 MOV DI,OFFSET NEXT
14 MOV CX,1024
15 REP MOVSB
16 MOV AH,1
17 INT 21H
18 MOV AH,4CH
19 INT 21H
20 CODE ENDS
21 END START

```

```
DOS
BOX DOSBox 0.74-3, Cpu speed: 3000 cycles, F...
-Q
C:\WORK>DEBUG TZ.EXE
-T
AX=0788 BX=0000 CX=0819 DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=0778 ES=0778 SS=0787 CS=0808 IP=0003  NU UP EI PL NZ NA PO NC
0808:0003 8ED8      MOV     DS,AX
-T
AX=0788 BX=0000 CX=0819 DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=0788 ES=0778 SS=0787 CS=0808 IP=0005  NU UP EI PL NZ NA PO NC
0808:0005 FC       CLD
-T
AX=0788 BX=0000 CX=0819 DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=0788 ES=0778 SS=0787 CS=0808 IP=0006  NU UP EI PL NZ NA PO NC
0808:0006 BE0000   MOV     SI,0000
-T
AX=0788 BX=0000 CX=0819 DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=0788 ES=0778 SS=0787 CS=0808 IP=0009  NU UP EI PL NZ NA PO NC
0808:0009 BF0004   MOV     DI,0400
-A
```

```
DOS
BOX DOSBox 0.74-3, Cpu speed: 3000 cycles, F...
0788:0320 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
0788:0330 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
0788:0340 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
0788:0350 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
0788:0360 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
0788:0370 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
-D
0788:0380 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
0788:0390 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
0788:03A0 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
0788:03B0 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
0788:03C0 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
0788:03D0 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
0788:03E0 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
0788:03F0 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
-D
0788:0400 00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0788:0410 00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0788:0420 00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0788:0430 00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0788:0440 00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0788:0450 00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0788:0460 00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0788:0470 00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
- ^_
```

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, F...
0788:0620 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
0788:0630 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
0788:0640 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
0788:0650 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
0788:0660 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
0788:0670 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
-D
0788:0680 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
0788:0690 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
0788:06A0 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
0788:06B0 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
0788:06C0 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
0788:06D0 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
0788:06E0 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
0788:06F0 46 46 46 46 46 46 46 46 46-46 46 46 46 46 46 46 FFFFFFFF
-D
0788:0700 00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0788:0710 00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0788:0720 00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0788:0730 00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0788:0740 00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0788:0750 00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0788:0760 00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0788:0770 00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
-A_
```

3.6 在以 DATA1 为首地址的主存区域中顺序存放着以 ASCII 码表示的十进制十位数，现欲将其转换成二进制数，试编写程序实现之。

3.7 试编写程序，将 MOLT 存储单元中的一个 8 位二进制数乘以 20，乘积放在 ANS 存储单元及下一单元中(用 3 种方法完成)。

3.8 在以 DATA 为首地址的主存区域中存放 100 个无符号 8 位数，试编写程序找出其

```
D:\DOS_Box\Work\work\T3.ASM - Sublime Text (UNREGISTERED)
文件(F) 编辑(E) 选择(S) 查找(F) 视图(V) 跳转(G) 工具(T) 项目(P) 首选项(N) 帮助(H)

practico1.html x practico2.html x test3.asm x test2.asm x TEST4.ASM x t2.asm x T3.ASM x t1.asm x
1 DATA SEGMENT
2 MOLT DB 01H
3 ANS DW ?
4 DATA ENDS
5
6 CODE SEGMENT ;程序段
7 ASSUME CS:CODE,DS:DATA
8 START:
9 MOV AX,DATA
10 MOV DS,AX
11
12 MOV AL,MOLT;第一种 al = al*20
13 MOV BL,20
14 MUL BL
15 MOV ANS,AX
16
17 MOV ANS,0
18
19 MOV AL,MOLT;第二种al = al*20 = al*(4*4+4)
20 MOV AH,0
21 MOV CL,2
22 SHL AX,CL
23 MOV BX,AX
24 SHL AX,CL
25 ADD AX,BX
26 MOV ANS,AX
27
28 MOV ANS,0
29
30 MOV AL,MOLT;第三种al = al*20 =al+al+al+....+al
31 MOV AH,0
```

```

32     MOV BX,0
33     MOV CX,20
34     PROCESS:ADD BX,AX
35     LOOP PROCESS
36     MOV ANS,BX
37
38     CODE ENDS
39     END START
40

```

DOSBox 0.74-3, Cpu speed: 3000 cycles, F...

Libraries [.lib]:
Definitions File [nul.def]:
LINK : warning L4021: no stack segment

C:\WORK>DEBUG T3.EXE
-T

AX=0788 BX=0000 CX=001F DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=0778 ES=0778 SS=0787 CS=0789 IP=0003 NU UP EI PL NZ NA PO NC
0789:0003 8ED8 MOV DS,AX
-T

AX=0788 BX=0000 CX=001F DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=0788 ES=0778 SS=0787 CS=0789 IP=0005 NU UP EI PL NZ NA PO NC
0789:0005 A00000 MOV AL,[0000] DS:0000=01
-D DS:0

0788:0000	01 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
0788:0010	B8 88 07 8E D8 A0 00 00	B3 14 F6 E3 A3 01 00 83
0788:0020	C4 0A E9 ED 01 B8 04 00	50 0E E8 D3 09 B8 73 27P.....s'
0788:0030	50 FF 76 E0 FF 76 DE B8	1A 53 50 0E E8 CB 07 B8	P.v.v...SP....
0788:0040	04 00 50 0E E8 B9 09 B8	6E 27 50 FF 76 F6 FF 76	..P.....n'P.v.v
0788:0050	F4 B8 22 53 50 0E E8 B1	07 B8 08 00 50 0E E8 9F	.."SP.....P...
0788:0060	09 FF 36 F6 56 FF 36 F4	56 B8 2A 53 50 0E E8 99	..6.U.6.U.*SP...
0788:0070	07 83 C4 1C 8B 46 EC 0B	46 EE 74 1D B8 04 00 50F..F.t....P

- ▲

DOSBox 0.74-3, Cpu speed: 3000 cycles, F...

-T

AX=0701 BX=0014 CX=001F DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=0788 ES=0778 SS=0787 CS=0789 IP=000A NU UP EI PL NZ NA PO NC
0789:000A F6E3 MUL BL
-T

AX=0014 BX=0014 CX=001F DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=0788 ES=0778 SS=0787 CS=0789 IP=000C NU UP EI PL NZ NA PO NC
0789:000C A30100 MOV [0001],AX DS:0001=0000
-T

AX=0014 BX=0014 CX=001F DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=0788 ES=0778 SS=0787 CS=0789 IP=000F NU UP EI PL NZ NA PO NC
0789:000F 83C40A ADD SP,+0A
-D DS:0

0788:0000	01 14 00 00 00 00 00 00	00 00 00 00 00 00 00 00
0788:0010	B8 88 07 8E D8 A0 00 00	B3 14 F6 E3 A3 01 00 83
0788:0020	C4 0A E9 ED 01 B8 04 00	50 0E E8 D3 09 B8 73 27P.....s'
0788:0030	50 FF 76 E0 FF 76 DE B8	1A 53 50 0E E8 CB 07 B8	P.v.v...SP....
0788:0040	04 00 50 0E E8 B9 09 B8	6E 27 50 FF 76 F6 FF 76	..P.....n'P.v.v
0788:0050	F4 B8 22 53 50 0E E8 B1	07 B8 08 00 50 0E E8 9F	.."SP.....P...
0788:0060	09 FF 36 F6 56 FF 36 F4	56 B8 2A 53 50 0E E8 99	..6.U.6.U.*SP...
0788:0070	07 83 C4 1C 8B 46 EC 0B	46 EE 74 1D B8 04 00 50F..F.t....P

- ▲

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, F...
-T
AX=0010 BX=0004 CX=0002 DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=0788 ES=0778 SS=0787 CS=0789 IP=0022  NU UP EI PL NZ AC PO NC
0789:0022 03C3      ADD     AX,BX
-T
AX=0014 BX=0004 CX=0002 DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=0788 ES=0778 SS=0787 CS=0789 IP=0024  NU UP EI PL NZ NA PE NC
0789:0024 A30100     MOV     [0001],AX      DS:0001=0000
-T
AX=0014 BX=0004 CX=0002 DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=0788 ES=0778 SS=0787 CS=0789 IP=0027  NU UP EI PL NZ NA PE NC
0789:0027 C70601000000 MOV     WORD PTR [0001],0000      DS:0001=0014
-D DS:0
0788:0000 01 14 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0788:0010 B8 88 07 8E D8 A0 00 00-B3 14 F6 E3 A3 01 00 C7 .....
0788:0020 06 01 00 00 00 A0 00 00-B4 00 B1 02 D3 E0 8B D8 .....
0788:0030 D3 E0 03 C3 A3 01 00 C7-06 01 00 00 00 A0 00 00 .....
0788:0040 B4 00 BB 00 00 B9 14 00-03 D8 E2 FC 89 1E 01 00 .....
0788:0050 F4 B8 22 53 50 0E E8 B1-07 B8 08 00 50 0E E8 9F .."SP.....P...
0788:0060 09 FF 36 F6 56 FF 36 F4-56 B8 2A 53 50 0E E8 99 ..6.U.6.U.*SP...
0788:0070 07 83 C4 1C 8B 46 EC 0B-46 EE 74 1D B8 04 00 50 .....F..F.t....P
-▲
```

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, F...
0789:003C 891E0100     MOV     [0001],BX      DS:0001=0000
-D DS:0
0788:0000 01 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0788:0010 B8 88 07 8E D8 A0 00 00-B3 14 F6 E3 A3 01 00 C7 .....
0788:0020 06 01 00 00 00 A0 00 00-B4 00 B1 02 D3 E0 8B D8 .....
0788:0030 D3 E0 03 C3 A3 01 00 C7-06 01 00 00 00 A0 00 00 .....
0788:0040 B4 00 BB 00 00 B9 14 00-03 D8 E2 FC 89 1E 01 00 .....
0788:0050 F4 B8 22 53 50 0E E8 B1-07 B8 08 00 50 0E E8 9F .."SP.....P...
0788:0060 09 FF 36 F6 56 FF 36 F4-56 B8 2A 53 50 0E E8 99 ..6.U.6.U.*SP...
0788:0070 07 83 C4 1C 8B 46 EC 0B-46 EE 74 1D B8 04 00 50 .....F..F.t....P
-T
AX=0001 BX=0014 CX=0000 DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=0788 ES=0778 SS=0787 CS=0789 IP=0040  NU UP EI PL NZ NA PE NC
0789:0040 F4      HLT
-D DS:0
0788:0000 01 14 00 00 00 00 00 00 00-00 00 00 00 00 00 .....
0788:0010 B8 88 07 8E D8 A0 00 00-B3 14 F6 E3 A3 01 00 C7 .....
0788:0020 06 01 00 00 00 A0 00 00-B4 00 B1 02 D3 E0 8B D8 .....
0788:0030 D3 E0 03 C3 A3 01 00 C7-06 01 00 00 00 A0 00 00 .....
0788:0040 B4 00 BB 00 00 B9 14 00-03 D8 E2 FC 89 1E 01 00 .....
0788:0050 F4 B8 22 53 50 0E E8 B1-07 B8 08 00 50 0E E8 9F .."SP.....P...
0788:0060 09 FF 36 F6 56 FF 36 F4-56 B8 2A 53 50 0E E8 99 ..6.U.6.U.*SP...
0788:0070 07 83 C4 1C 8B 46 EC 0B-46 EE 74 1D B8 04 00 50 .....F..F.t....P
-▲
```

3.10 在 BVFF 存储单元中有一个 BCD 数 A, 试编写程序计算 Y, 结果送 DES 存储单元。其中:

$$Y = \begin{cases} 3A, & A \leq 20 \\ A - 20, & 20 < A < 60 \\ 80, & A \geq 60 \end{cases}$$

3.11 在当前数据段(由 DS 决定)中偏移地址为 DATAB 开始的顺序 80 个存储单元里, 存放着某班 80 个同学某门考试的成绩。

文件(F) 编辑(E) 选择(S) 查找(I) 视图(V) 跳转(G) 工具(T) 项目(P) 首选项(N) 帮助(H)

```
practice1.html  practice2.html  test3.asm  tes
1  DATA SEGMENT
2      BVFF byte 21
3      DES DB ?
4  DATA ENDS
5
6  CODE SEGMENT
7      ASSUME CS:CODE,DS:DATA
8      START:
9      MOV AX,DATA
10     MOV DS,AX
11
12     MOV AL,BVFF
13     CMP AL,20
14     JL SELCET1
15     CMP AL,60
16     JL SELCET2
17     MOV AL,80
18     JMP RESULT
19     SELCET1:MOV BL,3
20     MUL BL
21     JMP RESULT
22     SELCET2:SUB AL,20
23     RESULT:MOV DES,AL
24 CODE ENDS
25 END START
26
```

原来为 07
21 为另一个测试数据
;程序段


```

DOSBox 0.74-3, Cpu speed: 3000 cycles, F...
Libraries [.lib]:
Definitions File [nul.def]:
LINK : warning L4021: no stack segment

C:\WORK>DEBUG T4.EXE
-T

AX=0788 BX=0000 CX=002F DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=0778 ES=0778 SS=0787 CS=0789 IP=0003  NV UP EI PL NZ NA PO NC
0789:0003 8ED8      MOV     DS,AX
-T

AX=0788 BX=0000 CX=002F DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=0788 ES=0778 SS=0787 CS=0789 IP=0005  NV UP EI PL NZ NA PO NC
0789:0005 A00000    MOV     AL,[0000]      DS:0000=07
-D DS:0
0788:0000 07 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0788:0010 B8 88 07 8E D8 A0 00 00 3C 14 7C 08 3C 3C 7C 0A .....<.!<<!.
0788:0020 B0 50 EB 08 B3 03 F6 E3-EB 02 2C 14 A2 01 00 27 ..P.....,....'
0788:0030 50 FF 76 E0 FF 76 DE B8-1A 53 50 0E E8 CB 07 B8 P.v..v...SP.....
0788:0040 04 00 50 0E E8 B9 09 B8-6E 27 50 FF 76 F6 FF 76 ..P.....n'P.v..v
0788:0050 F4 B8 22 53 50 0E E8 B1-07 B8 08 00 50 0E E8 9F .."SP.....P...
0788:0060 09 FF 36 F6 56 FF 36 F4-56 B8 2A 53 50 0E E8 99 ..6.U.6.U.*SP...
0788:0070 07 83 C4 1C 8B 46 EC 0B-46 EE 74 1D B8 04 00 50 .....F..F.t....P

```

```

DOSBox 0.74-3, Cpu speed: 3000 cycles, F...
AX=0015 BX=0003 CX=002F DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=0788 ES=0778 SS=0787 CS=0789 IP=001F  NV UP EI NG NZ NA PE NC
0789:001F 27      DAA
-T

AX=0015 BX=0003 CX=002F DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=0788 ES=0778 SS=0787 CS=0789 IP=0020  NV UP EI PL NZ NA PO NC
0789:0020 50      PUSH    AX
-T

AX=0015 BX=0003 CX=002F DX=0000 SP=FFFE BP=0000 SI=0000 DI=0000
DS=0788 ES=0778 SS=0787 CS=0789 IP=0021  NV UP EI PL NZ NA PO NC
0789:0021 FF76E0    PUSH    [BP-20]      SS:FFFE0=8926
-TD DS:0
^ Error
-D DS:0
0788:0000 07 15 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0788:0010 B8 88 07 8E D8 A0 00 00 3C 14 7C 08 3C 3C 7C 0A .....<.!<<!.
0788:0020 B0 50 EB 08 B3 03 F6 E3-EB 02 2C 14 A2 01 00 27 ..P.....,....'
0788:0030 50 FF 76 E0 FF 76 DE B8-1A 53 50 0E E8 CB 07 B8 P.v..v...SP.....
0788:0040 04 00 50 0E E8 B9 09 B8-6E 27 50 FF 76 F6 FF 76 ..P.....n'P.v..v
0788:0050 F4 B8 22 53 50 0E E8 B1-07 B8 08 00 50 0E E8 9F .."SP.....P...
0788:0060 09 FF 36 F6 56 FF 36 F4-56 B8 2A 53 50 0E E8 99 ..6.U.6.U.*SP...
0788:0070 07 83 C4 1C 8B 46 EC 0B-46 EE 74 1D B8 04 00 50 .....F..F.t....P
-A

```

3.13 试编写程序，给从主存 40000H 到 4BFFFH 的每个单元中均写入 55H，并逐个单元读出比较。若写入的与读出的完全一致，则将 AL 置 7EH；若有错，则将 AL 置 81H。

3.14 试编写程序，统计由主存 40000H 开始的 16K 个单元中所存放的字符“A”的个数，并将结果存放在 DX 中。

3.15 试采用 MMX 和 SSE2 技术分别编写程序实现两个 8×8 矩阵 A 和 B 的乘积，结

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, F...
0000:00A0 20 15 00 F0 40 15 00 F0-C0 15 00 F0 60 10 00 F0 ...@
0000:00B0 60 10 00 F0 60 10 00 F0-60 10 00 F0 A0 15 00 F0
0000:00C0 60 10 00 F0 60 10 00 F0-60 10 00 F0 10 00 FF C7
0000:00D0 60 10 00 F0 60 10 00 F0-60 10 00 F0 60 10 00 F0
0000:00E0 60 10 00 F0 60 10 00 F0-60 10 00 F0 60 10 00 F0
0000:00F0 60 10 00 F0 60 10 00 F0-60 10 00 F0 60 10 00 F0
-d
0000:0100 60 10 00 F0 60 11 00 F0-60 10 00 F0 00 17 00 C0
0000:0110 60 10 00 F0 60 10 00 F0-80 11 00 F0 60 10 00 F0
0000:0120 60 10 00 F0 60 10 00 F0-60 10 00 F0 00 16 00 F0
0000:0130 60 10 00 F0 60 10 00 F0-60 10 00 F0 60 10 00 F0
0000:0140 60 10 00 F0 60 10 00 F0-60 10 00 F0 60 10 00 F0
0000:0150 60 10 00 F0 60 10 00 F0-60 10 00 F0 60 10 00 F0
0000:0160 60 10 00 F0 60 10 00 F0-60 10 00 F0 60 10 00 F0
0000:0170 60 10 00 F0 60 10 00 F0-60 10 00 F0 60 10 00 F0
-d
0000:0180 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00
0000:0190 00 00 00 00 00 00 00-60 10 00 F0 04 00 41 C8
0000:01A0 60 10 00 F0 60 10 00 F0-60 10 00 F0 60 10 00 F0
0000:01B0 60 10 00 F0 60 10 00 F0-60 10 00 F0 60 10 00 F0
0000:01C0 80 12 00 F0 A0 12 00 F0-00 00 00 00 00 00 00
0000:01D0 20 14 00 F0 00 00 00 00-00 00 00 00 00 00 00
0000:01E0 00 00 00 00 00 00 00-00 00 00 00 00 00 00
0000:01F0 00 00 00 00 00 00 00-00 00 00 00 00 00 00
-d ▲
```

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, F...
AX=FF00 BX=0000 CX=3934 DX=0001 SP=0000 BP=0000 SI=06CC DI=0000
DS=0000 ES=0778 SS=0787 CS=0788 IP=000E NU UP EI PL NZ NA PE CY
0788:000E 8A04 MOV AL,[SI] DS:06CC=00

AX=FF00 BX=0000 CX=3934 DX=0001 SP=0000 BP=0000 SI=06CC DI=0000
DS=0000 ES=0778 SS=0787 CS=0788 IP=0010 NU UP EI PL NZ NA PE CY
0788:0010 3C41 CMP AL,41

AX=FF00 BX=0000 CX=3934 DX=0001 SP=0000 BP=0000 SI=06CC DI=0000
DS=0000 ES=0778 SS=0787 CS=0788 IP=0012 NU UP EI NG NZ AC PO CY
0788:0012 7501 JNZ 0015

AX=FF00 BX=0000 CX=3934 DX=0001 SP=0000 BP=0000 SI=06CC DI=0000
DS=0000 ES=0778 SS=0787 CS=0788 IP=0015 NU UP EI NG NZ AC PO CY
0788:0015 46 INC SI

AX=FF00 BX=0000 CX=3934 DX=0001 SP=0000 BP=0000 SI=06CD DI=0000
DS=0000 ES=0778 SS=0787 CS=0788 IP=0016 NU UP EI PL NZ NA PO CY
0788:0016 E2F6 LOOP 000E

AX=FF00 BX=0000 CX=3933 DX=0001 SP=0000 BP=0000 SI=06CD DI=0000
DS=0000 ES=0778 SS=0787 CS=0788 IP=000E NU UP EI PL NZ NA PO CY
0788:000E 8A04 MOV AL,[SI] DS:06CD=00
- ▲
```