

Lab 3

P1: Write a program to add an array of eight 2-digit hexadecimal numbers stored in memory and store the result in memory.

```

LAB 3
P1: .MODEL SMALL
    .STACK 20
    .DATA
    ORG 1000H
    NUM DB 25H,35H,45H,32H,56H,98H,76H,76H
    SUM DW ?
    COUNT DW 0008H
    .CODE
    START:
    MOV AX,@DATA
    MOV DS,AX
    MOV CX,COUNT
    MOV SI,0000H
    MOV AX,0000H
    REPEAT: ADD AL,NUM[SI]
            JNC NEXT
            ADD AH,01
    NEXT: INC SI
    LOOP REPEAT
    MOV SUM,AX
    INT 3
    END START
  
```

The screenshot shows the DOS DEBUG program interface. The CPU window displays the instruction stream and register values. The DS window shows the memory dump starting at address 1000H, where the array of eight 2-digit hexadecimal numbers is stored. The SI register points to the first element of the array at address 1000H.

Address	Instruction	Register/Value
cs:0000	mov ax,48AF	ax 02AB
cs:0003	mov ds,ax	bx 0000
cs:0005	mov cx,[100AH]	cx 0000
cs:0009	mov si,0000	dx 0000
cs:000C	mov ax,0000	si 0000
cs:000F	add al,[si+1000]	di 0000
cs:0013	jnb 0018	bp 0000
cs:0015	add ah,01	sp 0014
cs:0018	46	
cs:0019	E2F4	
cs:001B	A30810	
cs:001E	CC	
cs:001F	0000	

The DS window shows the memory dump starting at address 1000H:

Address	Value
ds:1000	25 35 45 32 56 98 76 76
ds:1008	AB 02 00 00 00 00 00 00
ds:1010	03 02 00 00 00 00 AF 48
ds:1018	DF 01 C5 15 AB 02 1F 00

The ES window shows the stack segment values:

Address	Value
es:0000	CD 20 FF 9F 00 EA FF FF
es:0008	AD DE E0 01 C5 15 AA 01
es:0010	C5 15 89 02 20 10 92 01
es:0018	01 03 01 00 02 FF FF FF

The SS window shows the stack segment values:

Address	Value
ss:0016	0000
ss:0014	0000

DS: 1000H

Output at 1008, 02AB (ie the sum of the numbers in array)

P2: Write a program to count number of occurrences of the byte 25H in the given array of 16-bytes stored starting from 1200H. Also store the result in 1220H memory location.

```

P2. .MODEL SMALL
    .STACK 20
    .DATA
        ORG 1200H
        ARRAY DB 25H, 35H, 45H, 32H, 56H, 25H,
                76H, 76H, 28H, 56H, 05H, 35H,
                25H, 00H, 98H, 21H

        ORG 1220H
        RES DB ?
        COUNT DW 0010H

    .CODE
    START:
        MOV AX, @DATA
        MOV DS, AX
        MOV CX, COUNT
        MOV SI, 0000H
        MOV AL, 25H
    REPEAT: CMP AL, ARRAY[SI]
            JNE NEXT
            INC RES
    NEXT: INC SI
        LOOP REPEAT
        INT 3
    END START

```

File Edit View Run Breakpoints Data Options Window Help

CPU 80486

cs:0000	B8AE48	mov	ax, 48AE	ax	4825	c=0
cs:0003	BED8	mov	ds, ax	bx	0000	z=0
cs:0005	8B0E2D12	mov	cx, [122D]	cx	0000	s=0
cs:0009	BE0000	mov	si, 0000	dx	0000	o=0
cs:000C	B025	mov	al, 25	si	0010	p=0
cs:000E	3A840C12	cmp	al, [si+120C]	di	0000	a=1
cs:0012	7504	jne	0018	bp	0000	i=1
cs:0014	FE062C12	inc	byte ptr [122]	sp	0014	d=0
cs:0018	46					

[I]=Dump 2=[111]

cs:0019	E2F3	ds:120C	25 35 45 32 56 25 76 76	%5E2U%w
cs:001B	CC	ds:1214	28 56 05 35 25 00 98 21	(Ux5% ij!
cs:001C	0000	ds:121C	00 00 00 00 00 00 00 00	
cs:001E	0000	ds:1224	00 00 00 00 00 00 00 00	

es:0000	CD 20 FF 9F 00 EA FF FF	= f R	
es:0008	AD DE E0 01 C5 15 AA 01	i R+S-R	
es:0010	C5 15 89 02 20 10 92 01	+Se R+R	ss:0016 0000
es:0018	01 03 01 00 02 FF FF FF	R+R R	ss:0014 0000

F1-Help F2-Bkpt F3-Mod F4-Here F5-Zoom F6-Next F7-Trace F8-Step F9-Run F10-Menu

DS:120C

File Edit View Run Breakpoints Data Options Window Help READY

CPU 80486 1

cs:0000 B8AE48	mov	ax, 48AE	ax	4825	c=0
cs:0003 8ED8	mov	ds, ax	bx	0000	z=0
cs:0005 8B0E2D12	mov	cx, [122D]	cx	0000	s=0
cs:0009 BE0000	mov	si, 0000	dx	0000	o=0
cs:000C B025	mov	al, 25	si	0010	p=0
cs:000E 3A840C12	cmp	al, [si+120C]	di	0000	a=1
cs:0012 7504	jne	0018	bp	0000	i=1
cs:0014 FE062C12	inc	byte ptr [122]	sp	0014	d=0

cs:0018 46 []=Dump 2=[↑][↓]

cs:0019 E2F3 ds:1214 28 56 05 35 25 00 98 21 (U%5% j!)

cs:001B CC ds:121C 00 00 00 00 00 00 00 00

cs:001C 0000 ds:1224 00 00 00 00 00 00 00 00

cs:001E 0000 ds:122C 03 10 00 00 03 48 00 00 H

es:0000 CD 20 FF 9F 00 EA FF FF = f R

es:0008 AD DE E0 01 C5 15 AA 01 x S

es:0010 C5 15 89 02 20 10 92 01 S A

es:0018 01 03 01 00 02 FF FF FF S S

ss:0016 0000

ss:0014 0000

F1-Help F2-Bkpt F3-Mod F4-Here F5-Zoom F6-Next F7-Trace F8-Step F9-Run F10-Menu

Output at 122C ie 3 times 25H occurs in ARRAY

P3: Write a program to exchange two data blocks of length 10-bytes stored in memory starting from 1200H and 1220H respectively.

```
P3: .MODEL SMALL
    .STACK 20
    .DATA
    ORG 1200H
    ARRAY1 DB 05H, 15H, 25H, 35H, 45H, 55H,
              65H, 75H, 85H, 95H
    ORG 1220H
    ARRAY2 DB 0A1H, 0A2H, 0A3H, 0A4H, 0A5H, 0A6H,
              0A7H, 0A8H, 0A9H, 0AAH
    COUNT DW 000AH

    .CODE
    START:
    MOV AX, @DATA
    MOV DS, AX
    MOV CX, COUNT
    MOV SI, 0000H
    REPEAT: MOV AL, ARRAY1[SI]
             XCHG AL, ARRAY2[SI]
             MOV ARRAY1[SI], AL
             INC SI
             LOOP REPEAT

    INT 3
    END START
```

```

≡ File Edit View Run Breakpoints Data Options Window Help READY
CPU 80486
cs:0000 B8AE48    mov     ax,48AE      ax 48AA  c=0
cs:0003 8ED8      mov     ds,ax        bx 0000  z=0
cs:0005 8B0E3612   mov     cx,[1236]    cx 0000  s=0
cs:0009 BE0000     mov     si,0000      dx 0000  o=0
cs:000C 8A840C12   mov     al,[si+120C] si 000A  p=1
cs:0010 86842C12   xchg    [si+122C],al di 0000  a=0
cs:0014 88840C12   mov     [si+120C],al bp 0000  i=1
cs:0018 46        inc     si        sp 0014  d=0
cs:0019 E2F1      [ ]=Dump          2=[1][1]
cs:001B>CC        ds:1204 00 00 00 00 00 00 00 00
cs:001C 0000      ds:120C A1 A2 A3 A4 A5 A6 A7 A8 1000N^=z
cs:001E 0000      ds:1214 A9 AA 00 00 00 00 00 00
cs:0020 0000      ds:121C 00 00 00 00 00 00 00 00

es:0000 CD 20 FF 9F 00 EA FF FF = f 0
es:0008 AD DE E0 01 C5 15 AA 01 1000S-0
es:0010 C5 15 89 02 20 10 92 01 1000S-0
es:0018 01 03 01 00 02 FF FF FF 0000
ss:0016 0000
ss:0014>0000

F1-Help F2-Bkpt F3-Mod F4-Here F5-Zoom F6-Next F7-Trace F8-Step F9-Run F10-Menu

```

```

≡ File Edit View Run Breakpoints Data Options Window Help READY
CPU 80486
cs:0000 B8AE48    mov     ax,48AE      ax 48AA  c=0
cs:0003 8ED8      mov     ds,ax        bx 0000  z=0
cs:0005 8B0E3612   mov     cx,[1236]    cx 0000  s=0
cs:0009 BE0000     mov     si,0000      dx 0000  o=0
cs:000C 8A840C12   mov     al,[si+120C] si 000A  p=1
cs:0010 86842C12   xchg    [si+122C],al di 0000  a=0
cs:0014 88840C12   mov     [si+120C],al bp 0000  i=1
cs:0018 46        inc     si        sp 0014  d=0
cs:0019 E2F1      [ ]=Dump          2=[1][1]
cs:001B>CC        ds:121C 00 00 00 00 00 00 00 00
cs:001C 0000      ds:1224 00 00 00 00 00 00 00 00
cs:001E 0000      ds:122C 05 15 25 35 45 55 65 75 5EUEu
cs:0020 0000      ds:1234 85 95 0A 00 DF 01 C5 15 0000S-0

es:0000 CD 20 FF 9F 00 EA FF FF = f 0
es:0008 AD DE E0 01 C5 15 AA 01 1000S-0
es:0010 C5 15 89 02 20 10 92 01 1000S-0
es:0018 01 03 01 00 02 FF FF FF 0000
ss:0016 0000
ss:0014>0000

F1-Help F2-Bkpt F3-Mod F4-Here F5-Zoom F6-Next F7-Trace F8-Step F9-Run F10-Menu

```

DS:120C

Values interchanged at 120C and 122C

P4: Write a program to arrange the given array of 8-bit binary numbers stored in the memory in ascending order. NUM DB 95H, 85H, 75H, 65H, 55H, 45H, 35H, 25H

```

P4: .MODEL SMALL
    .STACK 20
    .DATA
    ORG 1000H
    NUM DB 95H, 85H, 75H, 65H, 55H, 45H, 35H, 25H

    .CODE
    START:
        MOV AX, @DATA
        MOV DS, AX
        MOV CH, 07H

    X2: MOV CL, 07H
        LEA SI, NUM

    X1: MOV AL, [SI]
        MOV BL, [SI+1]
        CMP AL, BL
        JC REPEAT
        MOV DL, [SI+1]
        XCHG [SI], DL
        MOV [SI+1], DL

    REPEAT: INC SI
            DEC CL
            JNZ X1
            DEC CH
            JNZ X2
            INT 3
            END START
    
```

The screenshot shows the DOS DEBUG program interface. The menu bar at the top includes File, Edit, View, Run, Breakpoints, Data, Options, Window, and Help. The status bar at the bottom shows function key shortcuts: F1-Help, F2-Bkpt, F3-Mod, F4-Here, F5-Zoom, F6-Next, F7-Trace, F8-Step, F9-Run, and F10-Menu.

The main window is divided into several sections:

- Register Window (Top Right):** Displays the current state of CPU registers.

ax	4885	c	1
bx	0095	z	1
cx	0000	s	0
dx	0035	o	0
si	100F	p	1
di	0000	a	0
bp	0000	i	1
sp	0014	d	0
ds	48AF		
es	489D		
ss	49B0		
cs	48AD		
ip	0026		
- Command Window (Left):** Shows the assembly code being executed.


```

cs:0026 CC      int     03
cs:0027 0000     add     [bx+si],al
cs:0029 0000     add     [bx+si],al
cs:002B 0000     add     [bx+si],al
cs:002D 0000     add     [bx+si],al
cs:002F 0000     add     [bx+si],al
cs:0031 0000     add     [bx+si],al
cs:0033 0000     add     [bx+si],al
cs:0035 0000     add     [bx+si],al
cs:0037 0000     add     [bx+si],al
cs:0039 0000     add     [bx+si],al
cs:003B 0000     add     [bx+si],al
cs:003D 0000     add     [bx+si],al
            
```
- Memory Window (Bottom):** Displays the contents of memory locations.

ds:1000	00 00 00 00 00 00 00 00
ds:1008	25 35 45 55 65 75 85 95
ds:1010	03 48 00 00 35 00 AF 48
ds:1018	DF 01 C5 15 85 48 27 00

DS:1000H

Output at 1008H, numbers arranged in ascending order