

ASM - Lab Assignment 3



NAME : Swarnendu Banerjee

ROLL : 002311001016

DEPARTMENT : IT – A1

1. Write an Assembly Language Program to find the smallest number from a series of seven data bytes stored from DS: 0030H. Store the smallest number in DS: 0040H.

Code:

```
.model small
.stack 100h
.data
.code

main proc
mov ax, @data
mov ds, ax

mov cl, 07h
mov al, 00ffh
mov si, 0030h

l1:
cmp al,[si]
jc l2
mov al, [si]
l2: inc si
loop l1

mov si, 0040h
mov [si], al

int 03h
mov ah, 4ch
int 21h

main endp
end main
```

Output:

```

C:\>debug a3q1.exe
-t
AX=076C BX=0000 CX=001F DX=0000 SP=0100 BP=0000 SI=0000 DI=0000
DS=075A ES=075A SS=076C CS=076A IP=0003  NV UP EI PL NZ NA PO NC
076A:0003 8ED8          MOV     DS,AX
-e 076c:0030
076C:0030 01.1      00.2      02.3      00.4      03.5      00.6      04.7

-g=0000
AX=0701 BX=0000 CX=0000 DX=0000 SP=0100 BP=0000 SI=0040 DI=0000
DS=076C ES=075A SS=076C CS=076A IP=001A  NV UP EI PL NZ NA PO CY
076A:001A CC          INT     3
-d 076c:0040,0040
076C:0040 01

```

- Write an Assembly Language Program to find the largest number from a series of 7 sixteen-bit numbers stored from DS: 0030H. Store the largest number in DS: 0040H.

Code:

```

.model small
.stack 100h
.data
.code

main proc
mov ax, @data
mov ds, ax

mov cx, 0007h
mov ax, 0000h
mov si, 0030h

l1: cmp ax, [si]
jnc l2
mov ax, [si]
l2: add si, 02h
loop l1

mov si, 0040h
mov [si], ax

int 03h
mov ah, 4ch
int 21h

```

```
main endp
end main
```

Output:

```
C:\>debug a3q2.exe
-t
AX=076C BX=0000 CX=0023 DX=0000 SP=0100 BP=0000 SI=0000 DI=0000
DS=075A ES=075A SS=076D CS=076A IP=0003  NU UP EI PL NZ NA PO NC
076A:0003 8ED8          MOV     DS,AX
-e 076c:0030
076C:0030  00.01  01.00  00.02  02.00  00.03  03.00  00.04  04.00
-e 076c:0030
076C:0030  01.01  00.00  02.02  00.00  03.03  00.00  04.04  00.00
076C:0038  01.01  01.01  02.02  02.02  03.03  03.03
-g=0000
AX=0303 BX=0000 CX=0000 DX=0000 SP=0100 BP=0000 SI=0040 DI=0000
DS=076C ES=075A SS=076D CS=076A IP=001E  NU UP EI PL NZ NA PO NC
076A:001E CC          INT     3
-d 076c:0040,0041
076C:0040  03 03
```

3. Write an Assembly Language Program to arrange a series of 7 data bytes stored from DS: 0030H in ascending order.

Code:

```
.model small
.stack 100h
.data
.code

main proc
mov ax, @data
mov ds, ax

mov si, 0030h

mov bl, 06h

l3:
mov si, 0030h
```

```
mov cl, 06h
```

```
l1:
```

```
mov al, [si]
```

```
inc si
```

```
cmp al, [si]
```

```
jc l2
```

```
mov dl, [si]
```

```
mov [si], al
```

```
dec si
```

```
mov [si], dl
```

```
inc si
```

```
l2: loop l1
```

```
dec bl
```

```
cmp bl, 00h
```

```
jnz l3
```

```
int 03h
```

```
mov ah, 4ch
```

```
int 21h
```

```
main endp
```

```
end main
```

Output:

```
C:\>debug bubblesort.exe
-t

AX=076C  BX=0000  CX=0028  DX=0000  SP=0100  BP=0000  SI=0000  DI=0000
DS=075A  ES=075A  SS=076D  CS=076A  IP=0003  NU UP EI PL NZ NA PO NC
076A:0003 8ED8          MOV     DS,AX
-e 076c:0030
076C:0030 3D.02  FF.05  FF.01  74.03  03.06  E9.04  ED.07

-g=0000

AX=0706  BX=0000  CX=0000  DX=0000  SP=0100  BP=0000  SI=0036  DI=0000
DS=076C  ES=075A  SS=076D  CS=076A  IP=0023  NU UP EI PL ZR NA PE NC
076A:0023 CC          INT     3
-d 076c:0030,0036
076C:0030 01 02 03 04 05 06 07  ....
```

4. Write an Assembly Language Program to arrange a series of 7 sixteen-bits data stored from DS: 0030H in descending order.

Code:

```
.model small
.stack 100h
.data
.code

main proc
mov ax, @data
mov ds, ax

mov si, 0030h
mov bl, 06h

I3:
mov si, 0030h
mov cl, 06h

I1:
mov ax, [si]
add si, 02h
cmp ax, [si]
jnc I2
mov dx, [si]
mov [si], ax
sub si, 02h
mov [si], dx
add si, 02h

I2: loop I1
dec bl
cmp bl, 00h
jnz I3

int 03h
mov ah, 4ch
int 21h

main endp
end main
```

```
C:\>debug a3q4.exe
-t

AX=076D  BX=0000  CX=0032  DX=0000  SP=0100  BP=0000  SI=0000  DI=0000
DS=075A  ES=075A  SS=076E  CS=076A  IP=0003  NU UP EI PL NZ NA PO NC
076A:0003 8ED8          MOV     DS,AX
-e 076d:0030

076D:0030  E4.05    40.00    50.02    8B.00    C3.04    8C.00    C2.07    05.00
076D:0038  0C.06    00.00    52.01    50.00    E8.03    C1.00

-g=0000

AX=0002  BX=0000  CX=0000  DX=0006  SP=0100  BP=0000  SI=003C  DI=0000
DS=076D  ES=075A  SS=076E  CS=076A  IP=002D  NU UP EI PL ZR NA PE NC
076A:002D CC          INT     3
-d 076d:0030

076D:0030  07 00 06 00 05 00 04 00-03 00 02 00 01 00 48 83  .....
```

- Code:**

```
.model small
.stack 100h
.data
.code

main proc

    mov ax, @data
    mov ds, ax

    mov si, 0030h

;mov bl, [si]
;mov bh, 01h
;mov al, [bx]

    mov al, [si]
    mov bx, 0100h
    xlat
```

```
mov si, 0040h
mov [si], al
```

```
int 03h
mov ah, 4ch
int 21h
```

```
main endp
end main
```

Output:

```
C:\>debug assn3q5.exe
-t

AX=076B  BX=0000  CX=0018  DX=0000  SP=0100  BP=0000  SI=0000  DI=0000
DS=075A  ES=075A  SS=076C  CS=076A  IP=0003  NU UP EI PL NZ NA PO NC
076A:0003 8ED8          MOV     DS,AX
-e 076b:0100
076B:0100 FF.00  50.01  E8.04  89.09  69.16  83.25  6B.36  07.49
076B:0108 00.64  00.81

-e 076b:0030
076B:0030 00.05

-g=0000

AX=0725  BX=0100  CX=0018  DX=0000  SP=0100  BP=0000  SI=0040  DI=0000
DS=076B  ES=075A  SS=076C  CS=076A  IP=0013  NU UP EI PL NZ NA PO NC
076A:0013 CC          INT     3
-d 076b:0040,0040
076B:0040 25                                     %
```