

PROBLEM

Magnolia (WORD, low byte)	0x0072A302	0x12
Canvas (DWORD, low byte)	0x0072A303	0xCD
<hr/>		
<h2>PROBLEM 2</h2>		
0014E201 EBX = 02EA0000 ECX = 00A71005 EDX = FFFFFFFF ESI = 00A71005 EDI = 00A71005 EIP = 00A71010 ESP = 030FFB50 EBP = 030FFB5C EFL =		
TITLE Addressing Modes	(main.asm)	
INCLUDE Irvine32.inc		
.data		
alpha	DWORD	0A5B5C5D5h, 8E376C7Ah
beta	DWORD	3ABED9D2h, 21A220C2h
gamma	DWORD	0DC71546Bh
.code		
main PROC		
mov eax, 5ABEFh;	Immediate	< 1ms elapsed
mov ecx, eax;	Register to Register	
mov edi, OFFSET beta;	Immediate	
mov [gamma], eax;	Direct	
mov esi, gamma;	Direct	
mov esi, 4;	Immediate	
mov eax, beta[esi];	Indirect-offset	

```
beta          DWORD  
gamma         DWORD  
.code  
  
main PROC  
    mov eax, 5ABEFh;  
    mov ecx, eax;  
    mov edi, OFFSET beta;  
    mov [gamma], eax;
```

```
19     mov eax, 4[ebx];           Indirect-displacement
20     mov eax, [ebx];           Indirect
21     mov eax,4[ebx][esi];     Base-Indirect-displacement
22
23     exit
24 END main

Registers
EAX = 0005ABEF EBX = 02EA0000 ECX = 0005ABEF EDX = FFFFFFFF ESI = 00A71005 EDI = 00A71005 EIP = 00A71017 ESP = 030FFB50 EBP = 030FFB5C EFL = 00000203
175 % - < main.asm < x
1 TITLE Addressing Modes          (main.asm)
2
3 INCLUDE Irvine32.inc
4 .data
5     alpha    DWORD      0A5B5C5D5h, 8E376C7Ah
6     beta     DWORD      3ABED9D2h, 21A220C2h
7     gamma    DWORD      0DC71546Bh
8 .code
9
10    main PROC
11        mov eax, 5ABEFh;       Immediate
12        mov ecx, eax;         Register to Register
13        mov edi, OFFSET beta; Immediate      ≤ 1ms elapsed
14        mov [gamma], eax;     Direct
15        mov esi, gamma;      Direct
16        mov esi, 4;           Immediate
17        mov eax, beta[esi];   Indirect-offset
18        mov ebx, OFFSET alpha; Immediate
19        mov eax, 4[ebx];       Indirect-displacement
20        mov eax, [ebx];       Indirect
21        mov eax,4[ebx][esi];  Base-Indirect-displacement
22
23     exit
24 END main

Registers
EAX = 0005ABEF EBX = 02EA0000 ECX = 0005ABEF EDX = FFFFFFFF ESI = 00A71005 EDI = 00A74008 EIP = 00A7101C ESP = 030FFB50 EBP = 030FFB5C EFL = 00000203
175 % - < main.asm < x
1 TITLE Addressing Modes          (main.asm)
2
3 INCLUDE Irvine32.inc
4 .data
5     alpha    DWORD      0A5B5C5D5h, 8E376C7Ah
6     beta     DWORD      3ABED9D2h, 21A220C2h
7     gamma    DWORD      0DC71546Bh
8 .code
9
10    main PROC
11        mov eax, 5ABEFh;       Immediate
12        mov ecx, eax;         Register to Register
13        mov edi, OFFSET beta; Immediate
14        mov [gamma], eax;     Direct      ≤ 1ms elapsed
15        mov esi, gamma;      Direct
16        mov esi, 4;           Immediate
17        mov eax, beta[esi];   Indirect-offset
18        mov ebx, OFFSET alpha; Immediate
19        mov eax, 4[ebx];       Indirect-displacement
20        mov eax, [ebx];       Indirect
21        mov eax,4[ebx][esi];  Base-Indirect-displacement
22
23     exit
24 END main

Registers
EAX = 0005ABEF EBX = 02EA0000 ECX = 0005ABEF EDX = FFFFFFFF ESI = 00A71005 EDI = 00A74008 EIP = 00A71021 ESP = 030FFB50 EBP = 030FFB5C EFL = 00000203
175 % - < main.asm < x
1 TITLE Addressing Modes          (main.asm)
2
3 INCLUDE Irvine32.inc
4 .data
5     alpha    DWORD      0A5B5C5D5h, 8E376C7Ah
6     beta     DWORD      3ABED9D2h, 21A220C2h
7     gamma    DWORD      0DC71546Bh
8 .code
9
10    main PROC
11        mov eax, 5ABEFh;       Immediate
12        mov ecx, eax;         Register to Register
13        mov edi, OFFSET beta; Immediate
14        mov [gamma], eax;     Direct
15        mov esi, gamma;      Direct
16        mov esi, 4;           Immediate      ≤ 1ms elapsed
17        mov eax, beta[esi];   Indirect-offset
18        mov ebx, OFFSET alpha; Immediate
19        mov eax, 4[ebx];       Indirect-displacement
20        mov eax, [ebx];       Indirect
21        mov eax,4[ebx][esi];  Base-Indirect-displacement
22
23     exit
24 END main

Registers
EAX = 0005ABEF EBX = 02EA0000 ECX = 0005ABEF EDX = FFFFFFFF ESI = 0005ABEF EDI = 00A74008 EIP = 00A71027 ESP = 030FFB50 EBP = 030FFB5C EFL = 00000203
175 % - < main.asm < x
1 TITLE Addressing Modes          (main.asm)
2
3 INCLUDE Irvine32.inc
4 .data
5     alpha    DWORD      0A5B5C5D5h, 8E376C7Ah
6     beta     DWORD      3ABED9D2h, 21A220C2h
7     gamma    DWORD      0DC71546Bh
8 .code
9
10    main PROC
11        mov eax, 5ABEFh;       Immediate
12        mov ecx, eax;         Register to Register
13        mov edi, OFFSET beta; Immediate
14        mov [gamma], eax;     Direct
15        mov esi, gamma;      Direct
16        mov esi, 4;           Immediate      ≤ 1ms elapsed
17        mov eax, beta[esi];   Indirect-offset
18        mov ebx, OFFSET alpha; Immediate
19        mov eax, 4[ebx];       Indirect-displacement
20        mov eax, [ebx];       Indirect
21        mov eax,4[ebx][esi];  Base-Indirect-displacement
22
23     exit
24 END main

Registers
EAX = 0005ABEF EBX = 02EA0000 ECX = 0005ABEF EDX = FFFFFFFF ESI = 00000004 EDI = 00A74008 EIP = 00A7102C ESP = 030FFB50 EBP = 030FFB5C EFL = 00000203
175 % - < main.asm < x
1 TITLE Addressing Modes          (main.asm)
2
3 INCLUDE Irvine32.inc
4 .data
5     alpha    DWORD      0A5B5C5D5h, 8E376C7Ah
6     beta     DWORD      3ABED9D2h, 21A220C2h
7     gamma    DWORD      0DC71546Bh
8 .code
```

```
Registers  
EAX = 21A220C2 EBX = 02EA0000 ECX = 0005ABEF EDX = FFFFFFFF ESI = 00000004 EDI = 00A74008 EIP = 00A71032 ESP = 030FFB50 EBP = 030FFB5C EFL = 00000203  
175% ← ×  
main.asm ×  
1 TITLE Addressing Modes (main.asm)  
2  
3 INCLUDE Irvine32.inc  
4 .data  
5     alpha    DWORD    0A5B5C5D5h, 8E376C7Ah  
6     beta     DWORD    3ABED9D2h, 21A220C2h  
7     gamma    DWORD    0DC71546Bh  
8 .code  
9  
10 main PROC  
11     mov eax, 5ABEFh;      Immediate  
12     mov ecx, eax;        Register to Register  
13     mov edi, OFFSET beta; Immediate  
14     mov [gamma], eax;    Direct  
15     mov esi, gamma;    Direct  
16     mov esi, 4;          Immediate  
17     mov eax, beta[esi];  Indirect-offset      ≤ 1ms elapsed  
18     mov ebx, OFFSET alpha; Immediate  
19     mov eax, 4[ebx];    Indirect-displacement  
20     mov eax, [ebx];    Indirect  
21     mov eax,4[ebx][esi]; Base-Indirect-displacement  
22 exit  
23 main ENDP  
24 END main  
  
Registers  
EAX = 21A220C2 EBX = 00A74000 ECX = 0005ABEF EDX = FFFFFFFF ESI = 00000004 EDI = 00A74008 EIP = 00A71037 ESP = 030FFB50 EBP = 030FFB5C EFL = 00000203  
175% ← ×  
main.asm ×  
1 TITLE Addressing Modes (main.asm)  
2  
3 INCLUDE Irvine32.inc  
4 .data  
5     alpha    DWORD    0A5B5C5D5h, 8E376C7Ah  
6     beta     DWORD    3ABED9D2h, 21A220C2h  
7     gamma    DWORD    0DC71546Bh  
8 .code  
9  
10 main PROC  
11     mov eax, 5ABEFh;      Immediate  
12     mov ecx, eax;        Register to Register  
13     mov edi, OFFSET beta; Immediate  
14     mov [gamma], eax;    Direct  
15     mov esi, gamma;    Direct  
16     mov esi, 4;          Immediate  
17     mov eax, beta[esi];  Indirect-offset  
18     mov ebx, OFFSET alpha; Immediate  
19     mov eax, 4[ebx];    Indirect-displacement      ≤ 1ms elapsed  
20     mov eax, [ebx];    Indirect  
21     mov eax,4[ebx][esi]; Base-Indirect-displacement  
22 exit  
23 main ENDP  
24 END main  
  
Registers  
EAX = 8E376C7A EBX = 00A74000 ECX = 0005ABEF EDX = FFFFFFFF ESI = 00000004 EDI = 00A74008 EIP = 00A7103A ESP = 030FFB50 EBP = 030FFB5C EFL = 00000203  
175% ← ×  
main.asm ×  
1 TITLE Addressing Modes (main.asm)  
2  
3 INCLUDE Irvine32.inc  
4 .data  
5     alpha    DWORD    0A5B5C5D5h, 8E376C7Ah  
6     beta     DWORD    3ABED9D2h, 21A220C2h  
7     gamma    DWORD    0DC71546Bh  
8 .code  
9  
10 main PROC  
11     mov eax, 5ABEFh;      Immediate  
12     mov ecx, eax;        Register to Register  
13     mov edi, OFFSET beta; Immediate  
14     mov [gamma], eax;    Direct  
15     mov esi, gamma;    Direct  
16     mov esi, 4;          Immediate  
17     mov eax, beta[esi];  Indirect-offset  
18     mov ebx, OFFSET alpha; Immediate  
19     mov eax, 4[ebx];    Indirect-displacement  
20     mov eax, [ebx];    Indirect      ≤ 1ms elapsed  
21     mov eax,4[ebx][esi]; Base-Indirect-displacement  
22 exit  
23 main ENDP  
24 END main  
  
Registers  
EAX = A5B5C5D5 EBX = 00A74000 ECX = 0005ABEF EDX = FFFFFFFF ESI = 00000004 EDI = 00A74008 EIP = 00A7103C ESP = 030FFB50 EBP = 030FFB5C EFL = 00000203  
175% ← ×  
main.asm ×  
1 TITLE Addressing Modes (main.asm)  
2  
3 INCLUDE Irvine32.inc  
4 .data  
5     alpha    DWORD    0A5B5C5D5h, 8E376C7Ah  
6     beta     DWORD    3ABED9D2h, 21A220C2h  
7     gamma    DWORD    0DC71546Bh  
8 .code  
9  
10 main PROC  
11     mov eax, 5ABEFh;      Immediate  
12     mov ecx, eax;        Register to Register  
13     mov edi, OFFSET beta; Immediate  
14     mov [gamma], eax;    Direct  
15     mov esi, gamma;    Direct  
16     mov esi, 4;          Immediate  
17     mov eax, beta[esi];  Indirect-offset  
18     mov ebx, OFFSET alpha; Immediate  
19     mov eax, 4[ebx];    Indirect-displacement  
20     mov eax, [ebx];    Indirect  
21     mov eax,4[ebx][esi]; Base-Indirect-displacement      ≤ 1ms elapsed  
22 exit  
23 main ENDP  
24 END main
```

PROBLEM 2


```
05ABEF EBX = 02EA0000 ECX = 0005ABE  


---

TITLE Addressing Modes  
  
INCLUDE Irvine32.inc  
.data  
alpha DWORD 0A5B
```

MEMORY ADDRESS COMPUTATIONS

1. `mov [gamma], eax`

EA = 0A740010h

2. `mov esi, [gamma]`

EA = 0A740010h

3. `mov eax, beta[esi]` (with ESI = 4)

EA = $0A740008h + 4 = \underline{0A74000Ch}$

4. `mov eax, 4[ebx]` (with EBX = 0A740000h)

EA = EBX + 4 = 0A740004h

5. `mov eax, [ebx]` (with EBX = 0A740000h)

EA = EBX = 0A740000h

6. `mov eax, [ebx+esi+4]` (with EBX = 0A740000h, ESI = 4)

EA = EBX + ESI + 4 = 0A740008h

PROBLEM 3

The screenshot shows the Microsoft Visual Studio IDE with two windows open. The left window is titled "main.asm" and contains assembly language code. The right window is titled "Microsoft Visual Studio Debug" and shows a list of integers.

main.asm:

```
1 TITLE Sum of Array (main.asm)
2 INCLUDE Irvine32.inc
3
4 .data
5     CloudArray SWORD 11, 24, 3, 4, 15, 2, 7, 8, 19
6
7 .code
8 main PROC
9
10    xor eax, eax
11    mov ecx, LENGTHOF CloudArray
12    mov esi, OFFSET CloudArray
13
14    displayAndSum:
15        movsx edx, SWORD PTR [esi]
16
17        push eax
18        mov eax, edx
19        call WriteDec      ; I use 'WriteDec' to avoid signing the integers.
20        call Crlf
21        pop eax
22
23        add eax, edx
24
25        add esi, TYPE SWORD
26
27        loop displayAndSum
28
29        call WriteDec      ; See above comment.
30        call Crlf
31
32        exit
```

Microsoft Visual Studio Debug:

11
24
3
4
15
2
7
8
19
93

C:\Users\yolo\source\repos\HW5\Debug
Press any key to close this window