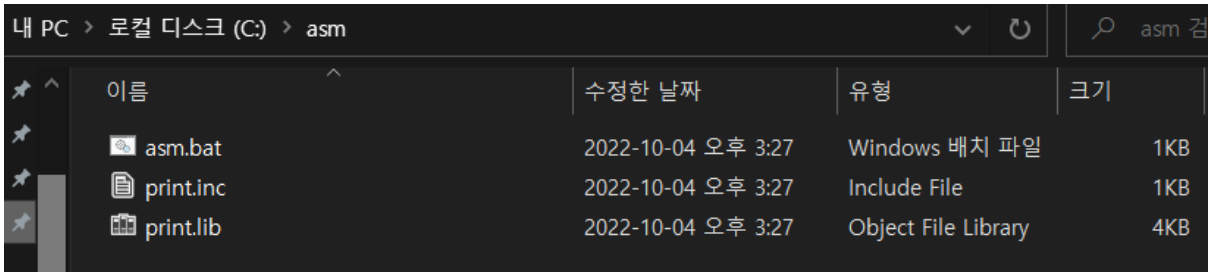
< 과제 1 >

2017253041_홍성우

1.

(1)



(2)

```
C:₩asm>notepad first2.asm

C:₩asm>asm first2

C:₩asm>REM asm.bat - batch file for assemble & link  assembly programs
Microsoft (R) Macro Assembler Version 14.27.29112.0
Copyright (C) Microsoft Corporation.  All rights reserved.

 Assembling: first2.asm
Microsoft (R) Incremental Linker Version 14.27.29112.0
Copyright (C) Microsoft Corporation.  All rights reserved.

C:₩asm>first2

C:₩asm>notepad first3.asm

C:₩asm>asm first3

C:₩asm>REM asm.bat - batch file for assemble & link  assembly programs
Microsoft (R) Macro Assembler Version 14.27.29112.0
Copyright (C) Microsoft Corporation.  All rights reserved.

 Assembling: first3.asm
Microsoft (R) Incremental Linker Version 14.27.29112.0
Copyright (C) Microsoft Corporation.  All rights reserved.

C:₩asm>first3

  EAX=0000000B  EBX=00C3D000  ECX=00AD1005  EDX=00AD1005
  ESI=00AD1005  EDI=00AD1005  EBP=00F7FF74  ESP=00F7FF68
  EIP=00AD1072  EFL=00000202  CF=0  SF=0  ZF=0  OF=0  AF=0  PF=0


C:₩asm>notepad first4.asm

C:₩asm>asm first4

C:₩asm>REM asm.bat - batch file for assemble & link  assembly programs
Microsoft (R) Macro Assembler Version 14.27.29112.0
Copyright (C) Microsoft Corporation.  All rights reserved.

 Assembling: first4.asm
Microsoft (R) Incremental Linker Version 14.27.29112.0
Copyright (C) Microsoft Corporation.  All rights reserved.

C:₩asm>first4
0000000B
C:₩asm>
```

first2는 출력결과가 없고, first3는 print.inc라는 파일을 include해 call dumpregs라는 프로시져로 레지스터의 값들을 보여주고 first4는 asm.bat에 print.lib를 포함하여 call writehex라는 프로시져로 해당 레지스터의 값을 보여준다.

(3)

```
C:₩asm>notepad second.asm

C:₩asm>asm second

C:₩asm>REM asm.bat - batch file for assemble & link  assembly programs
Microsoft (R) Macro Assembler Version 14.27.29112.0
Copyright (C) Microsoft Corporation.  All rights reserved.

 Assembling: second.asm
Microsoft (R) Incremental Linker Version 14.27.29112.0
Copyright (C) Microsoft Corporation.  All rights reserved.

C:₩asm>second

Dump of offset 00354000
-------------------------------
0000000B

C:₩asm>notepad second2.asm

C:₩asm>asm second2

C:₩asm>REM asm.bat - batch file for assemble & link  assembly programs
Microsoft (R) Macro Assembler Version 14.27.29112.0
Copyright (C) Microsoft Corporation.  All rights reserved.

 Assembling: second2.asm
Microsoft (R) Incremental Linker Version 14.27.29112.0
Copyright (C) Microsoft Corporation.  All rights reserved.

C:₩asm>second2

Dump of offset 00814000
-------------------------------
0000000B  00000003  00000005  00000007  00000009

C:₩asm>notepad second3.asm

C:₩asm>asm second3

C:₩asm>REM asm.bat - batch file for assemble & link  assembly programs
Microsoft (R) Macro Assembler Version 14.27.29112.0
Copyright (C) Microsoft Corporation.  All rights reserved.

 Assembling: second3.asm
Microsoft (R) Incremental Linker Version 14.27.29112.0
Copyright (C) Microsoft Corporation.  All rights reserved.
```

```
C:₩asm>second3
sum = 0000000B
Welcome

C:₩asm>
```

second는 dumpmem이라는 메모리 값 출력 프로시져를 사용하여 메모리 변수의 값을 출력했고 second2는 같은 방식이지만 첫번째 유닛에 eax값을 집어넣어서 변경된 값으로 출력했고 second3는 str1, str2의 각각 널문자가 종료된 문자열을 call writestring, call crlf이라는 프로시져를 사용하여 개행된 문자열을 출력했다.

(4)

```
C:₩asm>notepad endian.asm

C:₩asm>asm endian

C:₩asm>REM asm.bat - batch file for assemble & link  assembly programs
Microsoft (R) Macro Assembler Version 14.27.29112.0
Copyright (C) Microsoft Corporation.  All rights reserved.

 Assembling: endian.asm
Microsoft (R) Incremental Linker Version 14.27.29112.0
Copyright (C) Microsoft Corporation.  All rights reserved.

C:₩asm>endian

Dump of offset 00374000
--------------------------------
78 56 34 12

C:₩asm>
```

리틀엔디언 순서란 최하위 바이트가 최하위 메모리 주소에 저장되는 것을 뜻한다. 위 예시처럼 12345678h에서 최하위 바이트는 78이므로 메모리의 최하위 주소에 저장되고 나머지 123456은 78->56->34->12 순으로 저장된다.

(5)

```
C:₩asm>notepad equ.asm

C:₩asm>asm equ

C:₩asm>REM asm.bat - batch file for assemble & link  assembly programs
Microsoft (R) Macro Assembler Version 14.27.29112.0
Copyright (C) Microsoft Corporation.  All rights reserved.

 Assembling: equ.asm
Microsoft (R) Incremental Linker Version 14.27.29112.0
Copyright (C) Microsoft Corporation.  All rights reserved.

C:₩asm>equ

  EAX=00000064  EBX=00000006  ECX=00000069  EDX=00501005
  ESI=00501005  EDI=00501005  EBP=00AFFB3C  ESP=00AFFB30
  EIP=00501074  EFL=00000246  CF=0  SF=0  ZF=1  OF=0  AF=0  PF=1


C:₩asm>
```

기호상수를 정의하는 3가지 디렉티브는 =, equ, textequ이다. 위 예시처럼 matrix1은 정수식 10*10이고 matrix2는 matrix1과 같은 의미를 지닌 심볼형식이고 movcx의 textequ는 equ와 비슷하게 정수나 텍스트 형식으로 표현한 것이고 mov ecx, 105라는 의미로 사용되는 프로시져이다.

2.

(1)

```
C:₩asm>notepad mov.asm

C:₩asm>asm mov

C:₩asm>REM asm.bat - batch file for assemble & link  assembly programs
Microsoft (R) Macro Assembler Version 14.27.29112.0
Copyright (C) Microsoft Corporation.  All rights reserved.

 Assembling: mov.asm
Microsoft (R) Incremental Linker Version 14.27.29112.0
Copyright (C) Microsoft Corporation.  All rights reserved.

C:₩asm>mov

Dump of offset 00B84000
--------------------------------
FFFF8000  00008000

Dump of offset 00B84000
--------------------------------
00008000  FFFF8000

C:₩asm>
```

movsx는 sign 확장이고, movzx는 zero 확장이고, xchg는 dst와 src의 내용을 서로 교환한 것이다.

(2)

두 피연산자는 같은 크기여야 하고 메모리 간의 전송을 지원하지 않고 cs, eip는 dst로 사용 불가하고 즉시값은 segment register로 이동이 불가하다.

(3)

```
C:₩asm>notepad mov2.asm

C:₩asm>asm mov2

C:₩asm>REM asm.bat - batch file for assemble & link  assembly programs
Microsoft (R) Macro Assembler Version 14.27.29112.0
Copyright (C) Microsoft Corporation.  All rights reserved.

 Assembling: mov2.asm
Microsoft (R) Incremental Linker Version 14.27.29112.0
Copyright (C) Microsoft Corporation.  All rights reserved.

C:₩asm>mov2

Dump of offset 00844000
--------------------------------
00000040  00000010  00000020  00000030

C:₩asm>
```

mov2.asm - Windows 메모장

파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)

```
include print.inc

.data
array	dword	10h, 20h, 30h, 40h

.code
main proc
		mov	eax, array
		xchg	eax, array+4
		mov	array, eax

		mov	eax, array+8
		xchg	eax, array+12
		mov	array+8, eax

		mov	eax, array
		xchg	eax, array+8
		mov	array, eax

		mov	esi, offset array
		mov	ebx, 4
		mov	ecx, 4
		call	DumpMem

		mov	eax, 0
		call	ExitProcess
main endp
end main
```

Ln 1, Col 1 | 100% | Windows (CRLF) | UTF-8