

Chapter 5: Procedures(1)

Book's Library

The Book's Link Library

- Link Library Overview
- Calling a Library Procedure
- Linking to a Library
- Library Procedures – Overview
- Six Examples

Author's Link Library

■ Library file

- OBJ file들을 모아놓은 파일 (확장자 .LIB)
- 각 OBJ file에는 하나 이상의 procedure가 들어있음
- LIB 명령어를 사용하여 library file을 생성하며 OBJ file들의 추가, 삭제 가능

```
LIB user.lib +add.obj  
LIB user.lib -del.obj  
LIB user.lib -+ replace.obj
```

■ Author's library

- irvine32.lib 32-bit protected mode용 (make32로 어셈블)
- irvine16.lib 16-bit DOS 용 (make16으로 어셈블)
- 이 라이브러리에서 제공하는 함수에 대한 선언은 다음 파일에 포함
INCLUDE irvine32.inc
INCLUDE irvine16.inc

Calling a Library Procedure

■ Calling a Procedure

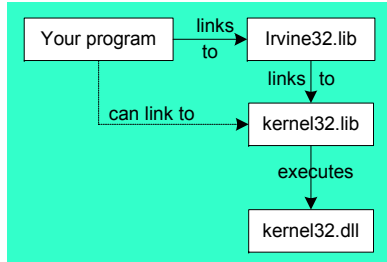
- CALL 명령어 사용
- 일부 procedure는 input argument를 필요로 함
 - argument는 register 또는 stack을 통해서 전달

■ 예: 화면에 1234를 출력

```
INCLUDE Irvine32.inc  
.code  
main proc  
    { mov eax,1234h          ; input argument  
      call WriteHex         ; show hex number  
      call Crlf             ; end of line  
    }  
main endp  
end main
```

Linking to a Library

- 사용자 프로그램을 library와 함께 링크하여 실행파일 생성
 - LINK32 *user.obj* *irvine32.lib* *kernel.lib* ; *make32.bat*에 포함됨
- *kernel32.lib*
 - MS-Windows OS에서 제공하는 *kernel32.dll*과의 bridge역할



Book's Libraries

| Procedure | Description |
|-----------------------|---|
| <i>Clrscr</i> | Clears the console (커서는 왼쪽 상단에 위치) |
| <i>CrLf</i> | Writes CR-LF (new line) |
| <i>Delay</i> | Pauses execution for a specified interval (단위 msec) (EAX: delay시간) |
| <i>DumpMem</i> | Writes a block of memory to standard output in hex. |
| <i>DumpRegs</i> | Displays the EAX, EBX, ECX, EDX, ESI, EDI, EBP, ESP, EFLAGS, and EIP registers in hex and flags(CF, SF, ZF, OF) |
| <i>GetCommandtail</i> | Copies the program's command-line arguments into an array of bytes. (EDX: buffer주소) |
| <i>GetMseconds</i> | Returns # of milliseconds that have elapsed since midnight. (EAX: 결과) |
| <i>Gotoxy</i> | Locates cursor at row and column on the console. (DL: X좌표, 0-79, DH: Y좌표, 0-24) |

Book's Libraries (계속)

| Procedure | Description |
|---------------------|--|
| <i>Random32</i> | Generates a 32-bit pseudorandom integer (EAX: 결과) |
| <i>Randomize</i> | Seeds the random number generator. (ECX: seed) |
| <i>RandomRange</i> | Generates a pseudorandom integer within a specified range (EAX: size입력, 결과 → range는 0 to size-1) |
| <i>ReadChar</i> | Reads a single character (AL: 결과) |
| <i>ReadHex</i> | Reads a 32-bit hex integer, terminated by Enter. (EAX: 결과) |
| <i>ReadInt</i> | Reads a 32-bit signed decimal integer, terminated by Enter. (EAX: 결과) |
| <i>ReadString</i> | Reads a string, terminated by Enter and insert a null terminator. (EDX: buffer주소, ECX: buffer의 크기) |
| <i>SetTextColor</i> | Sets the foreground and background colors of all subsequent text output to the console. (EAX: color) 하위4비트는 fgcolor(0-15), 상위4비트는 bgcolor |

Book's Libraries (계속)

| Procedure | Description |
|--------------------|--|
| <i>WaitMsg</i> | Displays message, waits for Enter key to be pressed. |
| <i>WriteBin</i> | Writes an unsigned 32-bit integer in binary format. (EAX: 정수) |
| <i>WriteChar</i> | Writes a single character. (AL: 문자) |
| <i>WriteDec</i> | Writes an unsigned 32-bit integer in decimal format. (EAX: 정수) |
| <i>WriteHex</i> | Writes an unsigned 32-bit integer in hex format. (EAX: 정수) |
| <i>WriteInt</i> | Writes a signed 32-bit integer in decimal format. (EAX: 정수) |
| <i>WriteString</i> | Writes a null-terminated string. (EDX: buffer주소) |

Example 1

- Clear the screen, delay the program for 500 msec, and dump the registers and flags

```
.code
main proc
    call Clrscr
    mov  eax,500
    call Delay
    call DumpRegs
main endp
end main
```

- Sample output

```
EAX=00000613 EBX=00000000 ECX=000000FF EDX=00000000
ESI=00000000 EDI=00000100 EBP=0000091E ESP=000000F6
EIP=00401026 EFL=00000286 CF=0 SF=1 ZF=0 OF=0
```

Example 2

- Display a null-terminated string and move the cursor to the beginning of the next screen line.

```
.data
str1 BYTE "Assembly language is easy!",0
.code
    mov  edx,OFFSET str1
    call WriteString
    call Crlf
```

```
; use embedded CR and LF
.data
str1 BYTE "Assembly language is easy!",0Dh,0Ah,0
.code
    mov  edx,OFFSET str1
    call WriteString
```

Example 3

- Display an unsigned integer in binary, decimal, and hexadecimal, each on a separate line.

```
IntVal = 35
.code
    mov  eax,IntVal
    call WriteBin          ; display binary
    call Crlf
    call WriteDec          ; display decimal
    call Crlf
    call WriteHex          ; display hexadecimal
    call Crlf
```

- Sample

```
0000 0000 0000 0000 0000 0000 0010 0011
35
23
```

Example 4

- Input a string from the user

```
.data
fileName BYTE 80 DUP(0)
.code
    mov  edx,OFFSET fileName
    mov  ecx,SIZEOF fileName - 1
    call ReadString
```

- A null byte is automatically appended to the string

Example 5

- Generate and display ten pseudorandom signed integers in the range 0 – 99.

```
.code
    mov  ecx,10          ; loop counter
L1: mov  eax,100          ; ceiling value
    call RandomRange     ; generate random int
    call WriteInt        ; display signed int
    call Crlf           ; goto next display line
    loop L1              ; repeat loop
```

← eax

Example 6

- Display a null-terminated string with yellow characters on a blue background
 - The background color is multiplied by 16 before being added to the foreground color.

```
.data
str1 BYTE "Color output is easy!",0
.code
    mov  eax, yellow + (blue * 16)
    call SetTextColor
    mov  edx, OFFSET str1
    call WriteString
    call Crlf
```