

# 과학적 표기법

## Scientific Notations

$$m \times 10^n$$

$m$  : significand (or mantissa)  
 $n$  : exponent

$$123.45 = 12345 \times 10^{-2} = 1.2345 \times 10^{+2}$$

유효숫자 5개  $1.2345 \times 10^{+2}$

유효숫자 6개  $1.2345\mathbf{0} \times 10^{+2}$

Modified normalized form :  $1.0 \leq m \leq 10$

True normalized form :  $0.1 \leq m \leq 1.0$

**Normalized significand** :  $1.\square\square\square\square \times 2^\Delta$

Normed significand :  $0.\square\square\square\square$

## 4 바이트(32비트) 부동소수점수



$$(+1) \times 2^{(124-127)} \times (1 + 2^{-2})$$

$$= +0.125 \times 1.25$$

$$= +0.15625$$

## 4 바이트 정수 범위

-2,147,483,648 ~ 2,147,438,647

대략  $-2.14 \times 10^9 \sim 2.14 \times 10^9$

## 4 바이트 부동소수점 범위

대략  $-3.4 \times 10^{38} \sim 3.4 \times 10^{38}$

10진수 유효숫자 6개

Navigation icons: back, forward, search, etc.

The screenshot shows the Visual Studio IDE with a C program being debugged. The code in the main function is as follows:

```
22 float f5 = 0xb.aP1;  
23 double d5 = 1.0625e0;  
24  
25 printf("%f %F %e %E\n", f, f, f, f);  
26 printf("%f %F %e %E\n", d, d, d, d);  
27 printf("%a %A\n", f5, f5);  
28 printf("%a %A\n", d5, d5);  
29  
30 return 0;  
31 }
```

The Solution Explorer on the right shows the project structure for 'Chapter3' (8 projects), including Lecture1 through Lecture9, and 'Lecture11.c'.

The Microsoft Visual Studio Debug Console shows the output of the program:

```
4  
8  
8  
123.456001 123.456001 1.234560e+02 1.234560E+02  
123.456000 123.456000 1.234560e+02 1.234560E+02  
0x1.7400000000000p+4 0X1.7400000000000P+4  
0x1.1000000000000p+0 0X1.1000000000000P+0  
D:\github-repository\TBC\Chapter3\Debug\Lecture11.exe (proce  
Press any key to close this window . . .
```