$a=10$

$b = ++a;$ $b=a+1$, $b=11 \rightarrow a=11$ ← a에 대입

$b = --a;$ $b=a-1$, $b=9 \rightarrow a=9$] 즉, $a=b$.

$b = a++;$ $b=10$, $a=b+1=11$

$b = a--;$ $b=10$, $a=b-1=9$] $a \neq b$

예제 4-2 (예측해보기)

$a=10$, $b=20$, $c=30$

$a = ++b;$ $a=20+1=21$, $b=21$.

$c = b++;$ $c=21$, $b=21+1=22$

$a = ++b + ++c;$

1을 각각의 변수에 더하고 계산할 것 같다.

$a = (1+22) + (1+21) = 45$, $b=23$, $c=22$

$a = b++ + c++;$

1을 더하기 전의 변수를 대입할 것 같다.

그후, 각각의 변수에 1을 더할 것 같다.

$a = 23+22$, $b = 23+1$, $c = 22+1$

$a=45$, $b=24$, $c=23$

예제 A-3 (예측)

```
int a=20, b=3, c=7, d, e ;
float f = 15.0, g;
```

d = a/b; // d = 20 ÷ 3 = 6
e = a%c; // e = 20 ÷ 7 의 (나머지) 20÷7 = 14···6
g = f/c; // g = 15.0 ÷ 7 인데   실수(of) 나깐
         소수점이 출력될 것 같다.
            2.0 (X)

출력: 2.142857
i) 소수점 6자리까지 계산된다.
ii) 반올림 하지 않는다.

논리 연산과

X	Y	AND (둘 다)	OR (하나라도)	NOT (거짓)
0	0	0	0	1
0	1	0	1	1
1	0	0	1	0
1	1	1	1	0

(X가 0 이면 1)
(1이면 0)

(&&)   (||)   (!X) !a, !b···

비트 연산

16진수	2진수
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
A	1010
B	1011
C	1100
D	1101
E	1110
F	1111

AND

3 & 5

```
      0011
AND   0101
    ------
      0001
```
⇒ 1

OR

3 | 5

```
       0011
OR     0101
     ------
       0111
```
⇒ 7

Shift 연산

14 >> 오른쪽으로 두 비트이동

1110 → 0011   [3]

3 << 왼쪽으로 두 비트이동

0011 → 1100   [12]