Rules:

1) only one var on left side of =

2) % optr,
   ↓
   only int
   sign = nmtr

3) on chars
   (ex) char x = 'a', y = 'b'
   int z = x + y
   Ascii
   a = 01100001 97
   b = 01100010 98

   3.1) optr should be explicitly mentioned

4) pow( )
   3^2 pow (3.0, 2.0)
   ↓only real values

conv
int → int
Real → Real
one int, one Real → always Real
and then proceeds.

ex) 5/2 = 2
5.0/2 = 2.5
5/2.0 = 2.5
5.0/2.0 = 2.5

Type conversions:

LHS = RHS
if is not same,

ex) float a,b,c;
int s;
s = a*b*c/100 + 32/4 - 3*1.1

k = 2/9 = 0	a = 2/9 = 0.000000
2.0/9 = 0	a = 2.0/9 = 0.222222
2/9.0 = 0	a = 2/9.0 ⇒ 0.222222
2.0/9.0 = 0	a = 2.0/9.0 "
9/2 = 4	a = 9/2 = 4.000000
9.0/2 = 4	a = 9.0/2 ⇒ 4.5
9/2.0 = 4	a = 9/2.0 ⇒ 4.5
9.0/2.0 = 4	a = 9.0/2.0 ⇒ 4.5
k = int	a = float

Bitwise

x 00001001
y 00010111

~x = 11110110
x & y = 00000001
x | y = 00011111
x ^ y = 00011110
x << 2 00100100
x >> 2

Format Specifiers

char | signed | unsigned → %c
int | short int | signed int | signed short int | unsigned short int → %d

Associativity : a = 3/2 * 5 (L → R)

a = b = 3 (R → L)

z = a*b + c/d

Hierarchy

*, /, %

+ -

=

++, --

( )

[ ]

++, --

+ , -

! ~

*

&

sizeof

*, /, %

+ -

<< >>

Logical

Bitwise AND

Logical AND

Conditional

Assignment

Comma

Paranthesis

ex) i = 2*3/4 + 4/4 + 8 - 2 + 5/8) evaluate this

unsigned int → %u

long int / signed long int → %ld

unsigned long int → %lu

double → %lf

Long double → %Lf

Prob: If a 5 digit number is input through the keyboard, write a C program to reverse the number

```
int n, d5, d4, d3, d2, d1;
int revnum;
d5 = n % 10;        d5 = 5
n = n / 10;         n = 1234
d4 = n % 10         d5 = 4
n = n / 10;         n = 123
d3 = n % 10         d3 = 3
n = n / 10          n = 12
d2 = n % 10         d2 = 2
n = n / 10          n = 1
d1 = n % 10         d1 = 1
revnum = d5 * 10000 + d4 * 1000
         + d3 * 100 + d2 * 10 + d1;
```