

このような演算子を代入演算子と呼びます．なお，今まで使っていた=も代入演算子です．

代入演算子の種類	
演算子と代入式	意味
$n_1 += n_2$	$n_1 + n_2$ を n_1 に代入
$n_1 -= n_2$	$n_1 - n_2$ を n_1 に代入
$n_1 *= n_2$	$n_1 * n_2$ を n_1 に代入
$n_1 /= n_2$	n_1 / n_2 を n_1 に代入
$n_1 \% = n_2$	n_1 / n_2 の剰余を n_1 に代入

1.5.4 繰り返し while

for 文は繰り返し回数が決まっている場合に使用しますが，while 文は繰り返し回数は決めずに条件式だけ与えられる場合に使用します．

```
while (条件式) {
    条件式が真のときの処理;
}
```

処理が 1 行で済む場合，ブロックを使わずに以下のようにも書けます．

```
while (条件式) 条件式が真のときの処理;
```

例えば，例題 9 では for 文を使い 2 の 16 乗まで計算しましたが，2 の何乗で 10,000 を超えるのかを知りたい場合など，ループ回数がわからない場合には while 文の方が適しています．

例題 10 while 文で 2 のべき乗を調べる

example-10.c

```
#include <stdio.h>
#include <math.h>

int main(void)
{
    int n=0, An=0, Sn=0;    /* n 項数, An 第 n 項, Sn 第 n 項までの和 */

    printf("n\t2 の n 乗\t数列の和\t\n");
    while (An < 10000) {
        n++;
        An = pow(2, n); /* 2 の n 乗を計算 */
        Sn += An;
        printf("%d\t%d\t%d\n", n, An, Sn);
    }
    printf("n = %d で 10,000 を超えます .\n", n);

    return 0;
}
```

実行結果

n	2 の n 乗	数列の和
1	2	2
2	4	6
3	8	14
4	16	30
5	32	62
6	64	126
7	128	254
8	256	510
9	512	1022
10	1024	2046
11	2048	4094
12	4096	8190
13	8192	16382
14	16384	32766

n = 14 で 10,000 を超えます .

1.5.5 繰り返し do while

while 文は条件式が偽の場合、処理を 1 度も行ないません。do while 文を使うとまず処理を 1 回行なってから条件式を評価します。

```
do {  
    条件式が真のときの処理;  
} while (条件式)
```

処理が 1 行で済む場合、ブロックを使わずに以下のようにも書けます。

```
do 条件式が真のときの処理; while (条件式);
```

例題 11 do while 文の使用例

example-11.c

```
#include <stdio.h>
int main(void)
{
    int i = 1;

    do {    /* 最低 1 度は実行される */
        printf("処理 1 i=%d\n", i);
        i++;

    } while (i < 0);

    do {    /* 最低 1 度は実行される */
        printf("処理 2 i=%d\n", i);
        i++;
    } while (i < 5);

    return 0;
}
```

実行結果

```
処理 1 i=1
処理 2 i=2
処理 2 i=3
処理 2 i=4
```

1.5.6 for と while の使い分け

for 文と while 文は書き方次第で同じ意味を持たせることができますが、用途によって使い分けるべきです。一般に繰り返し回数があらかじめわかっている場合は for 文、わかっていない場合は while 文で書けばいいでしょう。こうすると他の人が読んでも理解しやすいプログラムになります。

例題 12 for 文に向けた処理

example-12.c

```
#include <stdio.h>
int main(void)
{
    int i, n = 0;

    for (i = 1; i < 11; i++) {    /* 1 から 10 までの和を計算 */
        n += i;
    }
    printf("%d\n", n);
    return 0;
}
```

実行結果

55

例題 13 while 文に向いた処理

example-13.c

```
#include <stdio.h>
int main(void)
{
    char a;

    printf("A と入力するまで繰り返す %n");
    while ( a != 'A') { /* キーボードから A と入力されるまで繰り返す */
        scanf("%c", &a);
    }

    return 0;
}
```

実行結果

A と入力するまで繰り返す

a ← 以降 キーボードから入力

b

1

B

A

1.5.7 課題

課題 8 $S = \sum_{n=1}^{100} n$ を求めてみよう .

課題 9 例題 9 のプログラムを pow() 関数を使わずに書いてみよう .

課題 10 $e = \sum_{n=0}^{\infty} \frac{1}{n!}$ を求めてみよう . コンピュータでは計算を無限回繰り返すことはできないので適当に打ち切る .

課題 11 $f(x) = x^2$ の $f(x)$ を $-2 \leq x \leq 2$ の範囲で求めてみよう . x の刻み幅を 0.1 とする . 出力形式は次のようにする .

x	f(x)
-2.000000	4.000000
-1.900000	3.610000
-0.200000	0.040000
...	