第5章　「配列」のまとめ

5.1　用語のまとめ

教 p.110

配列　同一の型のデータを集めて、番号（＿＿＿、そえじ）でアクセスできるようにしたもの。C 言語の配列の添字は＿＿から始まる。

```
1   /* 初期化しないとき */
2   int va[5];
3   /* 配列の初期化は、式をコンマで区切って { } で囲む。*/
4   int vb[5] = { 15, 20, 30 };
5   /*（残りの要素は _で初期化される。）*/
6
7   /* 初期化子を代入することはできない。（教 p.114）*/
8   vb = {15, 20, 30, 0, 0 };
9   /* 配列同士の代入はできない（教 p.115）。*/
10  vb = va;
```

教 p.112

配列と for 文　配列は for 文と相性が良い。5 個の要素を持つ配列の各要素に対して同じ操作を行なうときには次のような for 文を使う。

```
for (______; _______; ____) {
   a[i] =  ... ;
}
```

教 p.117

配列を逆順に並びかえる　2 つの変数 x, y を入れ替えるのに、

```
x = y; y = x;
```

と書いてもダメで、別の変数（例えば temp）を一つ用意して、

＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿

と書く必要がある。

教 p.118

オブジェクト形式マクロ（定数マクロ）　プログラム中で繰り返し使う定数は名前をつける。

```
#define NUMBER 5
```

この指令は NUMBER というオブジェクト形式＿＿＿＿＿＿を定義する。マクロは他のコンパイル処理に先だって、一括して置換される。マクロを定義すると、次のような利点がある。

- 値の変更が容易になる。
- 定数の意味がわかり易くなる。秘密の数値（マジックナンバー）を直接プログラムに埋め込まないこと！

マクロが使われるのは、次のような箇所である。

- 配列の要素数など、文法上定数が要求されるところ
- 円周率などの数学定数・物理定数など絶対に変らない定数

（これら以外の箇所では、通常の変数を使うのが普通である。）

C99 規格では、配列の要素数に変数を使って、次のような書き方も可能になった。

```
int n = 3;
int a[n];  /* C99 では許容だが */

for (i = 0; i < n; i++) {
  a[i] = 0;
}
```

しかし、すべての処理系がこれをサポートしなくても良いことになっているので、非推奨とする。

マクロ名は通常すべての文字を＿＿＿＿＿とする慣習がある。小文字を使ってもコンパイルエラーになるわけではないが、強く非推奨とする。

教 p.121

代入演算子　代入（ 変数 = 式 ）も式であり、値（代入された値と同じ）を持つ。代入演算子は右結合である（右側から行われる）。つまり、`x = y = 0` は＿＿＿＿＿＿＿＿と解釈される。

教 p.124

多次元配列　各要素が配列であるような配列、言い替えれば 2 つ以上の添字を持つ配列のこと。ただし、物理的には一次元に配置される。（Fig.5-9 参照）

教 p.125

```
int x[2][3] = {{ 1, 2, 3 }, { 4, 5, 6 }};
```

Q 5.1.1 上の二次元配列 `x` のメモリ上の配置を Fig.5-9 のような図で表わせ。

5.2 プログラム例

教 p.97

break 文の例 (breakTest.c)

```
1   #include <stdio.h>
2   #define NUM 5
3
4   int main(void) {
5       int i;
6       int a[NUM] = {1, 2, -2, -4, 5};
7       for (i = 0; i < NUM; i++) {
8           if (a[i] < 0) {
9               break; /* continue; も試せ。 */
10          }
11          printf("a[%d] = %2d\n", i, a[i]);
12      }
13      return 0;
14  }
```

5.3 文法のまとめ

宣言 (declaration) に以下を追加する。

分類	一般形	補足説明
配列宣言	型 変数 [定数] = { 式 , ... , 式 } ;	= 以降の灰色の部分は省略可能

式 (expression) に以下を追加する。

分類	一般形	補足説明
配列アクセス	式 [式]	a[1], b[2][3] など