

第5章 「配列」のまとめ

5.1 用語のまとめ

教 p.88

配列 同じ型のデータを集めて、番号 (_____) 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。

```
int va[5]; /* 初期化しないとき */
int vb[5] = { 15, 20, 30 }; /* 配列の初期化は、式をコンマで区切って { } で囲む。
                             (残りの要素は 0 で初期化される。) */

vb = { 15, 20, 30, 0, 0 }; /* 初期化子を代入することはできない。(教 p.92) */
vb = va; /* 配列同士の代入はできない(教 p.93) */
```

教 p.90

配列と for 文 配列は for 文と相性が良い。配列の各要素に対して同じ操作を行なう時には for 文を使う。

```
for ( _____; _____; _____ ) {
    a[i] = ...;
}
```

教 p.94

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、別の変数を一つ用意して、

```
_____
```

と書く必要がある。

教 p.96

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
```

この前処理指令は `NUMBER` という _____ 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、

- 値の変更が容易になる。
- 定数の意味がわかり易くなる。(マジックナンバーを直接埋め込まないこと！)

マクロ名は通常 _____ にする。

教 p.98 代入演算子 代入 (変数 = 式) も式であり、値 (_____) を持つ。代入演算子は _____ である。つまり、`x = y = 0` は _____ と解釈される。

教 p.102 多次元配列 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。

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

教 p.104 **break 文** (もっとも内側の) 繰返し文 (`do ~ while 文`, `while 文`, `for 文`) を抜け出る。

continue 文 (参考) (もっとも内側の) 繰返し文のはじめ (`do ~ while 文`, `while 文` の場合は条件式、`for 文` の場合は第 3 式) にもどる。

break 文	continue 文

教 p.110 コンマ演算子
式₁, 式₂
は、式₁、式₂ をこの順に評価し、式₁ の値を捨て、式₂ の値 (と型) を持つ。

5.2 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
変数宣言	型 変数 = 式 , ... , 変数 = 式 ;	= 式 の部分は省略可能
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]	<code>a[1]</code> , <code>b[2][3]</code> など