

# 第7章 「基本型」のまとめ

## 7.1 用語のまとめ

### 整数型と文字型 (教 p.186)

`signed` は \_\_\_\_\_ (負の数も扱える)、`unsigned` は \_\_\_\_\_ (正の数と 0 のみ扱える) の整数を宣言する際の型指定子である。`short` と `long` はそれぞれ、標準よりも狭い、広い範囲を扱える（かもしれない）整数を宣言する際の型指定子である。`printf` 関数での `signed`, `long` に対する変換指定はそれぞれ \_\_\_\_\_, \_\_\_\_\_ である。

### limits.h ヘッダー (教 p.188)

各数値型で表現できる値の最小・最大値をマクロとして集めたヘッダーである。

**Q 7.1.1** 教科書 List 7-1 を実行し、次の空欄を埋めよ。

「 \_\_\_\_\_ の場合、`INT_MIN` は \_\_\_\_\_ 、`INT_MAX` は \_\_\_\_\_ 、`UINT_MAX` は \_\_\_\_\_ である。」

### sizeof 演算子 (教 p.192)

sizeof(型名)

という形で指定した型のサイズ（単位: バイト）を返す。

sizeof 式 /\* 通常は上の形にあわせて式を ( ~ ) で囲む \*/

という形で、式（通常は変数）のサイズ（単位: バイト）を返す。特に、式が配列の場合は配列全体のサイズ（単位: バイト）を返す。

**Q 7.1.2** 配列 `a` の要素数を表す sizeof 演算子を使った式は？

ただし、関数の引数として渡された配列では、`sizeof` 演算子は別の値（ポインター型のサイズ）を返すので注意する（プログラム例 `sizeof.c` 参照）。

### typedef 宣言 (教 p.194)

`typedef` 宣言は型の別名をつける。構造体・ポインターを学習したあとは頻繁に使う。

分類	一般形	補足説明
typedef 宣	<code>typedef 型 新しい型</code>	例えば <code>typedef unsigned</code>

言	名 ;	size_t;
---	-----	---------

### ビット演算 (教 p.202)

ビット単位の論理演算・シフト演算などは組込み用途では多用される。必要に応じて調べられるようにしておく。

### 整数定数 (教 p.210)

8進定数は先頭に「  」を、16進定数は先頭に「  」をつけて表記する。

10進	8進	16進
48	060	0x30
65	0101	0x41
97	0141	0x61

### 整数の表示 (教 p.212)

printf関数で整数を8進数または16進数で表示するためには、それぞれ、  
  ,    (A～Fを大文字にしたいときは  ) という書式指定を用いる。

### math.h ヘッダー (教 p.217)

sin, cos, tan, sqrt (square root — 平方根), exp, logなどの数学関数のプロトタイプ宣言が集められているヘッダーである。

gccの場合、math.hヘッダーの関数を使ったプログラムをコンパイルするときは、大抵 -lmオプションが必要である。

### 繰返しの制御 (教 p.218)

誤差が累積するので、繰返しを制御する変数に、できるだけ浮動小数点数型 (float, double) は使わない。

#### Q 7.1.3 次のプログラム（の断片）の出力はどうなるか？

```
1 printf("%.60f", 0.1);
```

### 演算子の一覧 (教 p.221)

優先順位や結合性をすべて覚える必要はないが、必要に応じて表を調べるように、どのような演算子があるかくらいは覚えておきたい。(Table 7-13)

## 7.2 プログラム例

### sizeof演算子の確認 (sizeof.c)

```
1 #include <stdio.h>
2
```

```
3 void foo(int x[]) {  
4     printf("sizeof(x) = %u\n", (unsigned) sizeof(x));  
5 }  
6  
7 int main(void) {  
8     int a[] = { 1, 2, 3 };  
9  
10    printf("sizeof(a) = %u\n", (unsigned) sizeof(a));  
11    foo(a);  
12    return 0;  
13 }
```

関数の引数として渡される配列に対する `sizeof` はポインター型（第 10 章）のサイズを返す。

---

