

第2章 「演算と型」のまとめ

2.1 用語のまとめ

演算子 (教 p.22^旧24^新)

とは、「+」、「-」、「*」、「/」のように演算の働きを持った記号のことである。(教科書 p.205^旧221^新にC言語のすべての演算子の表がある。)

_____とは、その演算の対象となる式(変数や定数など)のことである。

「/」演算子 (教 p.23^旧25^新)

整数 / 整数

という演算では、整数としての割算(小数点以下は _____)の結果が得られる。

整数 % 整数

では、_____ (余り) を求める。

Q 2.1.1 次の式の値は? ① $3 / 10 \dots$ _____ ② $11 \% 3 \dots$ _____

printf で「%」文字を表示 (教 p.23^旧25^新)

「%%」と2つ重ねることで「%」という文字そのものを出力することができる。
puts 関数では、「%」はそのまま出力される。

複数の変換指定 (教 p.25^旧27^新)

変換指定(「%d」など)が複数あるときは、第2実引数, 第3実引数, ...が順に対応する。

Q 2.1.2 次の式の出力は?

① `printf("%d/%d", 10, 24) ...` _____

② `printf("(%d, %d, %d)", 12, 34, 5) ...` _____

代入演算子 (教 p.27^旧29^新)

「=」演算子のことは(単純)代入演算子と呼ばれる。

式 (教 p.27^旧29^新)

変数や定数、それらを演算子で結合したものを _____ という。

式文 (教 p.27^旧29^新)

式のあとに「;」をつけて文にしたものを**式文**という。

() (教 p.29^旧30^新)

式のなかで先に演算する箇所を示すために丸括弧「()」を用いる。他の括弧（{ }, [] など）は別の用途があるため、この目的には使用できない。

double型 (教 p.29^旧31^新)

とは、いわゆる実数（正確には浮動小数点数）を扱うための型である。もちろん、実数と言っても精度には限界がある。

scanf関数とdouble型 (教 p.31^旧33^新)

実数を読み込むときは「%lf」を用いる。

```
1 scanf("____", &fx); /* キーボードから実数を変数 fxに読み込
```

変換指定のまとめ (教 p.31^旧33^新)

以下の表くらいは、暗記しておくこと。

	int	double
printf	%d	注
scanf	____	____

注: _____

整数定数と浮動小数点定数 (教 p.31)

5, 10などは整数定数、2.718などは小数点を含むものは浮動小数点定数と呼ばれる。基本的に整数定数はint型、浮動小数点定数はdouble型である。

型と演算 (教 p.32^旧34^新)

実数 / 実数

の演算では、切捨ては行わず、通常の割算が行われる。一方intとdoubleが混じっている場合、

整数 / 実数

や

実数 / 整数

の場合も、整数(int)型のオペランドが_____が行われて実数(double)型になり、double型の演算となる。

キャスト (**cast**) (教 p.34^旧36^新)

とは、明示的に_____することである。

(型) 式

という形で、「式」の値を「型」としての値に変換する。例えば、

```
1 int na, nb;
2 ...
3 ... (double) (na + nb) / 2 ...
```

では、double 型としての割算が行われる。(演算子の優先順位に注意する(教 p.205)。割算よりもキャストが先に行われる。)

なお、double から int へのキャスト

```
1 double x = -2.8;
2 ... (int)x ...
```

は切捨てになる。

Q 2.1.3 次の式の値は?

① (double)1 / 2 ... _____ ② (double) (1 / 2) ... _____

高度な変換指定 (教 p.36^旧38^新)

以下のような変換指定は必要に応じて調べれば良い。

説明	例	出力
桁数を揃える	printf("%3d", 1)	[1]
桁数を揃え先頭を 0 で埋める	printf("%03d", 1)	[001]
小数点以下の桁数を指定する	printf("%.3f", 3.1415926)	[3.142]
16 進数で表示する (小文字)	printf("%x", 127)	[7f]
16 進数で表示する (大文字)	printf("%X", 127)	[7F]

Q 2.1.4 次の printf 関数の呼出しの出力は?

① printf("%.4f", 1.0 / 3) ... _____
② printf("%x", 32) ... _____

2.2 文法のまとめ

式 (**expression**)

に以下を追加、

分類	一般形	補足説明
単項演算	単項演算子 式	単項演算子は +, -, &, ... など
二項演算	式 二項演算子 式	二項演算子は +, -, *, /, =, ... など
カッコ	(式)	演算の順番を指定するため、

		括弧で囲んだもの (教 p.28)
浮動小数点 数 リテラル		3.14, 2.0, 6.02e23, 6.6626e-34 など (教 p.27)
キャスト	(型)式	明示的な型変換 (教 p.31)

文 (**statement**) とは

これまでのところ、

分類	一般形	補足説明
式文	式	式は通常、代入式か関数呼出し (教 p.27)