

補助資料 #10 (後半)

以下に教材の各段階ごとに、教室で説明する時に押さえているポイントなどについてコメントします。この補助資料と、元の教材を並列に見ながら少しずつ作業するようにしてください。

■ 配列を引き数に与える

教科書の当該部分を読んで、配列変数をまるごと関数の引数に使えることを把握してください。受け渡し方と、受け取った側での扱い方を理解して下さい。また、教材にもありますが、配列を引き数で渡した場合、関数内で値を修正すると呼び出し元でも反映される点に注意してください。

□ 課題 2. 最大の数を求める関数 (#4 教材の課題 2. から)

#4 当時のファイルを見つけるのが難しい人は、教材ページに [配列中の最大値をさがす array_max.c] としてサンプルがありますので、それを関数化してください。

□ 二次元配列を引き数に渡す

次の魔方陣判定をするタイミングではこの方法を利用しているのでスキップせず読んで把握するように。

□ 課題 3. 魔方陣判定関数

教材ページに [魔方陣判定関数 (ひな形) matrix_msq_skel.c] として、とりあえず入力を受け付けて二次元配列に格納し、isMagicSquare() 関数にその配列を引き渡すところまでのコードがあります。

これをまず手元に持ってきてコンパイル・実行し、以下のような表示が出る場所まで確認してください。

```
$ echo 1 2 3 4 5 6 7 8 9 | ./a.out
Not Magic Square
$
```

それから、判定関数の中身を書いて、教材 PDF の「注意」にあるように「右の判定例を試してすべてその通りになることを確認」しながら、つまりテストしながら開発してください。

テストを楽にするために、[魔方陣チェックスクリプト msq_check.sh] を教材に用意してあります。これをダウンロードして、以下のようにして実行してください。(コンパイルして作った実行ファイルが a.out であることを仮定しています。)

```
$ bash msq_check.sh ./a.out
===== ./a.out
##### Magic #####
Magic Square (sum=15)

##### NotMagic #####
Not Magic Square
$
```

このチェックスクリプトは (中身を見ると分かるでしょうが)、8 回、指定されたプログラムを、それぞれ異なるデータを与えて実行します。最初のデータが魔方陣となるもの。あとのデータは皆魔方陣とならないデータです。つまりあなたが作ったプログラムが正しく動作するなら、このチェックスクリプトを実行したとき、上のように 1 つの “Magic Square” 判定と、7 つの “Not Magic Square” 判定を表示するはずですが、そうならない (特に魔方陣とならないはずのデータを与えられたのに魔方陣であると誤判定する) ようであれば、あなたはそのプログラムを見直す必要があります。

見直す方法は単純で、以前にやった「ハンドトレース」を試みるのが良いです。