

# プログラミング演習 II

## 課題 4 — 関数 2 —

### 関数と配列の要点

```
int func( int x[], int num )
int 型や double 型の場合、配列とその要素数を渡す
int func( char str[] )
char 型 (文字列) の場合、文字数は不要
```

### 再帰関数呼出しの例 (階乗)

```
int fact( int n ) {
    if ( n>0 ) return (n*fact(n-1));
    else return (1); /* 停止条件 */
}
```

### プログラミング技術 : Level 8

1. プログラミング技術 Level 7 を満たしている。
2. 再利用を考慮した関数作りができる。
3. どのような単位で関数を作れば良いかを理解している。
4. main 関数の引数の与え方を理解している。

【課題】関数を用いて、以下の問題を解きなさい。ただし、大域変数を使用しないこと。

配列のサイズなど、定数は必ず `#define` を利用すること。

表示や入力を目的とする関数以外では、関数の中で標準入力や標準出力への入出力は行わないこと。

【練習問題】この問題は採点の対象外です。

練習 (a). main 関数を次のように宣言すると、実行時に引数を取れるようになる。引数の和を求めるプログラムを作成しなさい。

```
#include <stdio.h>
#include <stdlib.h>
int main( int argc, char *argv[] ) {
    /** argc: 引数の個数、argv[0]: プログラム名、argv[1] ~ argv[n]: 文字列 **/

    /* 文字列を実数に変換する */
    x = atof( argv[1] ); /* atof の使用例 */
    ...
    return(0);
}
```

問題 1. 任意の個数のデータに対して、その最大値を求める関数 `maxof()` を作成しなさい。ただし、データは `main` 関数への引数で与えるものとする。すなわち、`main` 関数を次のように宣言し、実行時に引数を取れるようにする。例えば、`./prog 1 2 3` と実行した場合は、3 と表示される。`main` 関数では、入力されたデータを配列に入れ、関数 `maxof()` を呼び出す。また、返却値で渡された最大値を表示する。

問題 2. (This question will be provided from Adrien sen-sei.)

問題 3. アルファベットから成る文字列 `str`(平文と言う) を JIS コード表にしたがい  $n$  文字ずつずらして暗号文 `cryptogram` を作成する関数を作成しなさい。このような暗号化をシーザー暗号 (Caesar cipher) と言う。JIS コード表の `0x21 (!)` から `0x7e (~)` までの文字全体を考える。空白文字 `0x20` はそのまま表示する (無変換)。平文と  $n$  を引数にとり、暗号文を返す関数を作りなさい。実行例には、平文を暗号化した場合と、その暗号文を平文に戻した場合を含めること。例えば、 $n = 5$  文字ずらす場合、次の表のようになる。従って、平文 "Cat zoo" は暗号文 "Hfy !tt" となる。

平 文	A	B	C	...	Z	...	,	a	...	z
暗号文	F	G	H	...	_	...	e	f	...	!

実行例

- $n$  は負の数も可能である。また、大きな数も取り得る。
- $n = 0x5d = 93$  のとき、元の文字と同じになる。
- 16 文字シフトすると、JIS コード表の隣りの列になる。

問題 4. フィボナッチ数列を再帰関数呼出しを用いて求める関数 `Fibonacci()` を作りなさい。フィボナッチ数列  $F_0, F_1, \dots, F_n$  は

$$F_0 = 0, \quad (1)$$

$$F_1 = 1, \quad (2)$$

$$F_i = F_{i-1} + F_{i-2}, \quad i \geq 2 \quad (3)$$

で定義される数列である。第  $n$  項までのフィボナッチ数列をすべて表示しなさい。ただし、関数 `Fibonacci(i)` は  $i$  番目の値  $F_i$  を返却する関数であり、内部では表示しない。`main` 関数から  $n$  回呼び出すこと。`long int` 型を使うともう少し大きな整数まで計算可能になる。このとき、`printf` の書式も変更すること。