

第 10 回課題 T10

(関数と再帰:教科書第 13 章、2007/06/21(木))

基本問題

T10-1:フィボナッチ数列

(本提出期限 2007/06/21(木)17:40、再提出期限 2007/7/5(木)14:30)

提出物: Makefile、ソースファイル (fibonacci.c)、入力ファイル (fibonacci.in)、出力ファイル (fibonacci.out)

フィボナッチ数列 F_n は非負整数 (0 以上の整数) に対し次のように帰納的 (再帰的) に定義される数列である。

$$F_n = \begin{cases} 0 & n = 0 \text{ のとき} \\ 1 & n = 1 \text{ のとき} \\ F_{n-1} + F_{n-2} & n > 1 \text{ のとき} \end{cases}$$

フィボナッチ数列の第 n 項 F_n の値を求めるプログラムを作成せよ。ただし、下の実行例を参考に、後ろに示す要求仕様を満たすように作成せよ。

実行例 1

```
b08b0xx@tyy:~/T10/1$ ./fibonacci
フィボナッチ数列 F の第 n 項 Fn を求めます。
n?
10
F10=          55
b08b0xx@tyy:~/T10/1$
```

実行例 2

```
b08b0xx@tyy:~/T10/1$ ./fibonacci
フィボナッチ数列 F の第 n 項 Fn を求めます。
n?
-3
n には 0 以上の整数を入力して下さい。
b08b0xx@tyy:~/T10/1$
```

要求仕様 (T10-1)

全体的な仕様:

- フィボナッチ数列の第 n 項 F_n の値を求める再帰的な関数 `fib` を作れ。関数 `fib` の仕様は後の説明を参照。
- 関数 `fib` を呼ぶ際には、関数 `fib` の仮引数の条件に合うように適切に実引数を設定する。
- 入力は標準入力から求めたい項番号 n (0 以上の整数, `int` 型で非負の値) を受け取る。
- 出力は、フィボナッチ数列の第 n 項 F_n の値を標準出力に行なう。入力される項番号 n が負のときには、適切なエラーメッセージを標準出力に行なう。

フィボナッチ数列の第 n 項を求める再帰的関数 `fib` の仕様:

- 次のプロトタイプ宣言を持つ。

```
/*フィボナッチ数列を求める再帰的関数*/
int fib(int n);
```
- 仮引数 n はフィボナッチ数列の求めたい項番号とし、0 以上の整数が与えられるとする。呼び出された段階で常に $n \geq 0$ が満たされるとして良く、関数 `fib` 中でエラー処理を行なう必要はない。
- 戻り値は、引数 n に与えられた項番号 n のフィボナッチ数列 F_n の値とする。戻り値は、0 以上の整数 (`int` 型の値) である。
- 関数 `fib` 中では、再帰呼出を用いること。すなわち、関数 `fib` の定義中で、関数 `fib` を用いること。
- 関数 `fib` 中では、`for` 文や `while` 文などによる繰り返しを利用せずに作成すること。
- 再帰的関数 `fib` 中では、基礎を `if` 文により明示すること。
- 関数 `fib` 中では、標準入出力による入出力は行なってはならない。例えば、`printf` 文による表示や、`scanf` 文による読み込みは行なわない。

応用問題

T10-2:再帰互除法

(本提出期限 2007/6/28(木)14:30、再提出期限 2007/7/5(木)14:30)

提出物 : Makefile、ソースファイル (rec_gcd.c)、入力ファイル (rec_gcd.in)、
出力ファイル (rec_gcd.out)

ユークリッドの互除法を利用して二つの自然数 p, q の最大公約数を求める
プログラムを作成せよ。ただし、下の実行例を参考に、後ろに示す要求仕様
を満たすように作成せよ。

実行例 1

```
b08b0xx@tyy:~/T10/2$ ./rec_gcd
最大公約数を求めます。
1 つの正の整数を入力して下さい。
1029
2 つめの正の整数を入力して下さい。
2688
 1029 と 2688 の最大公約数は 21 です。
b08b0xx@tyy:~/T10/2$
```

実行例 2

```
b08b0xx@tyy:~/T10/2$ ./rec_gcd
最大公約数を求めます。
1 つの正の整数を入力して下さい。
-1234
2 つめの正の整数を入力して下さい。
2688
正の整数ではないので、
最大公約数はありません。
b08b0xx@tyy:~/T10/2$
```

要求仕様 (T10-2)

全体的な仕様:

- ユークリッドの互除法を用いて最大公約数の値を求める再帰的な関数 `rec_euclid` を作れ。関数 `rec_euclid` の仕様は後の説明を参照。
- 関数 `rec_euclid` の仮引数の制約を満たすように適切に実定数を設定し、関数 `rec_euclid` を呼び出す。
- 入力には以下を満たすようにすべて標準入力から行なう。
 - 2 個の自然数 (`int` 型で 0 より大きい値) を標準入力から入力する。(問題中では) 最初に読み込まれる自然数を p とし、2 番目に読み込まれる自然数を q とする。
 - 自然数 p と自然数 q はどちらが大きいても良い。すなわち、入力順序を問わない。
- 出力は、以下を満たすようにすべて標準出力へ行なう。
 - 自然数 p と自然数 q の最大公約数を標準出力へ行なう。
 - 自然数 p あるいは自然数 q が 0 以下の場合には、最大公約数が存在しないため、適切なメッセージを標準出力へ出力する。

ユークリッドの互除法を用いて最大公約数を求める再帰的関数 `rec_euclid` の仕様:

- 次のプロトタイプ宣言を持つ .

```
/*ユークリッドの互除法により最大公約数を求める再帰的  
関数*/  
int rec_euclid(int dividend,int divisor);
```

- 各引数は次の意味と制約を持つ .
 - 引数 `dividend` は割られる数 (被除数) を意味し, 自然数 (1 以上の整数) が与えられるとする .
 - 引数 `divisor` は割る数 (除数) を意味し, 自然数 (1 以上の整数) が与えられるとする .
 - 被除数 (引数 `dividend` に与えられた値) は, 除数 (引数 `divisor` に与えられた値) 以上とする .
 - 呼び出された段階で, 上の 3 つの制約は常に満たされるとして良く, 関数 `rec_euclid` 内でエラー処理を行なう必要はない .
- 戻り値は, 被除数 (引数 `dividend` と除数 (引数 `divisor` に与えられた値) の最大公約数の値 (自然数) とする .
- 関数 `rec_euclid` 中では, 再帰呼び出しを用いること . すなわち, 関数 `rec_euclid` の定義中で, 関数 `rec_euclid` を利用すること .
- 再帰的関数 `rec_euclid` 中では, 基礎を `if` 文により明示すること .
- 関数 `rec_euclid` 中では, `for` 文や `while` 文などによる繰り返しを利用せずに作成すること .
- 関数 `rec_euclid` 中では, 標準入出力による入出力は行なってはならない . 例えば, `printf` 文による表示や, `scanf` 文による読み込みは行なわない .