

プロジェクトオイラーと J のプログラムの練習 (01-10)

SHIMURA Masato

2017 年 5 月 23 日

目次

| | | |
|----|--------------------------------------|----|
| 1 | Multiple of 3 and 5–Problem 1 | 3 |
| 2 | Even Fibonacci numbers-problem 2 | 4 |
| 3 | Largest Prime factor problem 3 | 5 |
| 4 | Largest palindrome product-problem 4 | 6 |
| 5 | Smallest Multiple-problem 5 | 8 |
| 6 | Sum of square difference-problem 6 | 11 |
| 7 | 10001st prime-problem 7 | 12 |
| 8 | Largest product in series-problem 8 | 13 |
| 9 | Specil phytagorean triplet-problem 9 | 15 |
| 10 | Problem 10 | 18 |

はじめに

この稿はゲームマニアの 2 人の身近な子たちに数学とプログラムへの誘いのために書いたものです。

プログラムの世界を広く見るための良い企画なので最初の部分をまとめてみました。

J の入門としては J の高級電卓機能だけでも解が求められるものもあるが、簡単な関数をループで回すものもあります。

正解後に、フォーラムで他の言語で書いた腕自慢の多様なプログラムも見

られます。

解は示すなどのことなのでミニマムな例題の解法にした。

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| o | o | o | o | o | o | o | o | o | o | o | | | o | o | o |

プロジェクトオイラー

数論とプログラムの腕試しの場がある。

<http://projecteuler.net>

最初の問題が 2004 年となっており、世界各国の数学好きの腕自慢 65 万人が手練れのプログラム言語を引っ提げて参戦している。問題は現在 600 問近くになっており、正解するとステータスポイントとランキングが与えられる。現在トップは日本人の Python 使いである。

ユーザー登録を済ませるとプロジェクトに参加できる。過去問は [Progress] に入っており、最初の 100 問は入門用となっている。回答は公表されていないので、今からでも第一問から参加でき、現在も参加者が多くいる。

正解すると、その問いのフォーラムを見ることができ、各種言語のプログラムが腕自慢により投稿されているので、色々な言語とプログラム手法や回答者の癖も見ることができる。正解率はかなり低いく 10 パーセントを割る。

使用言語は Python がトップで C 連合 (C/C++,Java ,C#) とが双耳峰をなしており、あとは実に多様な言語がなだらかな裾野を形成し、眺めるだけでも楽しい。

APL/J/K も数は多くないがステータスは高い。参加者の中には Pencil/Paper や Assembler,COBOL という猛者もいる。

*1

*1 スクリプトは J でのアルゴリズムの理解のために作成し、計算は途中で留めた。

1 Multiple of 3 and 5—Problem 1

If we list all the natural numbers below 10 that are multiples of 3 or 5, we get 3, 5, 6 and 9. The sum of these multiples is 23.

Find the sum of all the multiples of 3 or 5 below 1000.

10 より小さい 3 と 5 の自然数の倍数は 3,5,6,9 でその合計は 23 である。

3 と 5 の倍数で 1000 より小さいものの合計を求めよ

1.1 Script の例

- `10 <./ 3 5 */ >:i.10`

```
3 6 9 10 10 10 10 10 10
```

```
5 10 10 10 10 10 10 10 10
```

- 抜き出しの指標

```
ind=. 10>/ a=. 3 5 */ >:i.10
```

```
1 1 1 0 0 0 0 0 0
```

```
1 0 0 0 0 0 0 0 0
```

```
ind # /("1) a
```

```
3 6 9
```

```
5 0 0
```

```
+;/ ind # /("1) a
```

```
23
```

- 重複を除く (`~.`)

2 Even Fibonacci numbers-problem 2

Each new term in the Fibonacci sequence is generated by adding the previous two terms.

By starting with 1 and 2, the first 10 terms will be:

1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ...

By considering the terms in the Fibonacci sequence whose values do not exceed four million, find the sum of the even-valued terms.

フィボナッチ数は 1, 2 のように前の 2 数を加えるもので、最初の 10 数の合計は上のようになる。

フィボナッチ数で 400 万を超えないものの内、偶数の数の合計を求める

2.1 Script

```
fm=: 0 1 ,: 1 1x
```

```
mp=: +/ . *
```

```
fib=: fm & mp
```

```
NB. Usage: fib ^:(i.12) 1 1
```

```
pick_even=:3 : 0
```

```
NB. Usage: pick_even {. "1 fib ^:(i.10) 1 1
```

```
ind=. ({:"1) 2 x: 1r2* y) e.1
```

```
ind # y
```

```
)
```

2.2 経過と解説

- Fibonacci 用のマトリクス. 浮動小数点を避けるため拡張実数用の x を付加する

$$\begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}$$

- `mp=: +/ . *` 内積
- `fib=: fm & mp` フィボナッチ関数のタシット定義

- `^(i.10)` タシット (陰関数定義) でのループ
- 偶数の抜き出しは意外と難しい。1r2 をかけて、(2 x:) で整数部と小数部を分離して小数部の 1 を指標とする

2.3 実行例

```
fib ^:(i.33) 1 1
```

```

1      1
1      2
2      3
3      5
5      8
8     13
13    21
21    34
34    55
55    89
89   144
```

```
pick_even {:"1 fib ^:(i.20) 1 1
2 8 34 144 610 2584 10946
```

3 Largest Prime factor problem 3

The prime factors of 13195 are 5, 7, 13 and 29.

What is the largest prime factor of the number 600851475143 ?

13195 の素因数は 5,7,13,29

600851475143 の素因数で一番大きいものは

3.1 q: キューコロン

素因数は J の原始関数 q: で求められる。

q: 13195

5 7 13 29

4 Largest palindrome product-problem 4

A palindromic number reads the same both ways. The largest palindrome made from the product of two 2-digit numbers is $9009 = 91 \times 99$.

Find the largest palindrome made from the product of two 3-digit numbers.

回文数 (前、後から読んで同じ数) で 2 桁同士の最大は $91 \times 99 = 9009$

3 桁の回文数で最大のものを求める

- 一つの数の回文

以前に回文数のスクリプトを作成した (JAPLA 2012/09) が、それは 1 つの数をひっくり返して足す方式。今回は 2 数をかけて回文になる数での最大値を求める問題

285+582

867+768

867+768

1635+5361

1635+5361

6996

- リバース/reverse

```
reverse=: 3 : 0
NB. reverse 4567 --> 7654
Num=:# ": y
10&#. |.(Num # 10) #: y
)
```

```
kaibun_sub0=: 3 : ' y + reverse y '
```

- 回文では数の反転が肝要。文字化した後、基底を用いて数を分離して、反転後再度数に合成している
- reverse を用いて、回文数か否かの判定関数を作成する

```
check_kaibun=: 3 : 'y = reverse y '
```

```
NB. Usage u 9009
```

```
NB. return is 1/0 =yes/no
```

- 2 数の全ての組み合わせ catalog { を用いる

ボックス内の2数の掛け算

{a;a=. >:i.5

*/ L:0 {a;a=. >:i.5

```

+----+----+----+----+----+
| 1 1|1 2|1 3|1 4|1 5|
+----+----+----+----+----+
| 2 1|2 2|2 3|2 4|2 5|
+----+----+----+----+----+
| 3 1|3 2|3 3|3 4|3 5|
+----+----+----+----+----+
| 4 1|4 2|4 3|4 4|4 5|
+----+----+----+----+----+
| 5 1|5 2|5 3|5 4|5 5|
+----+----+----+----+----+

```

```

+----+----+----+----+
| 1|2 |3 |4 |5 |
+----+----+----+----+
| 2|4 |6 |8 |10|
+----+----+----+----+
| 3|6 |9 |12|15|
+----+----+----+----+
| 4|8 |12|16|20|
+----+----+----+----+
| 5|10|15|20|25|
+----+----+----+----+

```

- 4問の Forum に J で 1 行で書いた手練がいた。

5 Smallest Multiple-problem 5

2520 is the smallest number that can be divided by each of the numbers from 1 to 10 without any remainder.

What is the smallest positive number that is evenly divisible by all of the numbers from 1 to 20?

2520 は 1,2...10 で割り切れる最小の数

では 1 から 20 で割り切れる最小の数は

5.1 solution

1. 先ず頭に浮かぶのは力業法だが、今回はなかなか手強い。
2. J の GCD(+) LCM(*) は両項である。

```
(>:i.10) */ table >:i.10
+----+-----+
|*/./| 1  2  3  4  5  6  7  8  9 10|
+----+-----+
| 1 | 1  2  3  4  5  6  7  8  9 10|
| 2 | 2  2  6  4 10  6 14  8 18 10|
| 3 | 3  6  3 12 15  6 21 24  9 30|
| 4 | 4  4 12  4 20 12 28  8 36 20|
| 5 | 5 10 15 20  5 30 35 40 45 10|
| 6 | 6  6  6 12 30  6 42 24 18 30|
| 7 | 7 14 21 28 35 42  7 56 63 70|
| 8 | 8  8 24  8 40 24 56  8 72 40|
| 9 | 9 18  9 36 45 18 63 72  9 90|
|10 |10 10 30 20 10 30 70 40 90 10|
+----+-----+
```

3. グループ全体の LCM を求めるには

```
*/ >:i.10 NB. 全部掛け合わせる
3628800
```

```
q: 3628800 NB. 因数分解
2 2 2 2 2 2 2 2 3 3 3 3 5 5 7
```

```
*/ 2 3 5 7 NB. 重複排除のうえ掛け合わせる
```

210

```
210 % >:i.10x NB. 箱に入らないものがある
210 105 70 105r2 42 35 30 105r4 70r3 21
```

```
12* 210 % >:i.10x NB. はみ出た分母の大小公倍数をかける
2520 1260 840 630 504 420 360 315 280 252
```

```
12*210
2520
```

4. グループの LCM を求める関数を作る

```
LCM_group=: 3 : 0
NB. usage: LCM_group >: i. 10x //using x
tmp1=. */tmp0=: ~. q: */ y
tmp2=. }. sort_nub {:( "1) 2 x: tmp1 % y
tmp1;tmp2;tmp3;tmp1 * tmp3=. >./; */~ tmp2
)
```

```
sort_nub=: ~.@/:~
```

5. 実行

```
LCM_group >: i.10x
+----+-----+---+-----+
|210|2 3 4|12|2520|
+----+-----+---+-----+
```

6. 2 3 4 から 12 を求めるには公約数を倍数の大きい方に吸い寄せる作業が必要
対角行列は自乗なので不要

```

*/~ 2 3 4
4 6 8
6 9 12
8 12 16

```

7. 次から 24 を得るには工夫が必要

```

*/~ 2 3 4 8
4 6 8 16
6 9 12 24
8 12 16 32
16 24 32 64

```

```

*./ 2 3 4 8
24

```

8. ここまでやって、やっと気付いた。J の規格外の能力と簡潔すぎる辞書。
J の VOCABULARY に書いておいてほしい。

```

*./ >:i.10
2520

```

6 Sum of square difference-problem 6

The sum of the squares of the first ten natural numbers is,

$$1^2 + 2^2 + \dots + 10^2 = 385$$

The square of the sum of the first ten natural numbers is,

$$(1 + 2 + \dots + 10)^2 = 55^2 = 3025$$

Hence the difference between the sum of the squares of the first ten natural numbers and the square of the sum is $3025 - 385 = 2640$.

Find the difference between the sum of the squares of the first one hundred natural numbers and the square of the sum.

1...100 の 2 乗の合計と、合計の 2 乗の差を求める

- 1..10 の 2 乗とその合計

```

*:>: i.10
1 4 9 16 25 36 49 64 81 100
+/: *>: i.10 NB. sum <-----quare
385

```

- 1..10 の合計とその 2 乗

```

+/: >:i.10
55
*:/ >:i.10 NB. square <-----sum
3025

```

7 10001st prime-problem 7

By listing the first six prime numbers: 2, 3, 5, 7, 11, and 13, we can see that the 6th prime is 13.

What is the 10,001st prime number?

最初の 6 個の素数は 2,3,5,7,11,13 で、6 番目は 13

(1 は含まない)

10001 番目の素数は

- 最初から 10 個の素数 (1 は含まず)

```

p: i.10
2 3 5 7 11 13 17 19 23 29

```

- 0...5 番目を抽出

```

p: 5
13

```

8 Largest product in series-problem 8

The four adjacent digits in the 1000-digit number that have the greatest product are $9 \times 9 \times 8 \times 9 = 5832$.

73167176531330624919225119674426574742355349194934

96983520312774506326239578318016984801869478851843

85861560789112949495459501737958331952853208805511

12540698747158523863050715693290963295227443043557

.....

71636269561882670428252483600823257530420752963450

(progres より copy してください)

Find the thirteen adjacent digits in the 1000-digit number that have the greatest product.

What is the value of this product?

1000 桁の数の内で 4 の連続する数の積の最大値は $9 \times 8 \times 9 \times 9 = 5832$

13 の連続数の合計の最大値は

- 10 進への数値化

```
(9#10)#: 234567890
2 3 4 5 6 7 8 9 0
```

- オーバーラップしながら 4 個ずつに区切る

```
4<\i.10
+-----+-----+-----+-----+-----+-----+
|0 1 2 3|1 2 3 4|2 3 4 5|3 4 5 6|4 5 6 7|5 6 7 8|6 7 8 9|
+-----+-----+-----+-----+-----+-----+
```

- ボックス内の合計は */ L:0

- おまけ

```
>./ ;*/ (L:0) 13<\ (1000 # 10)#: p8
```

9 Specil pythagorean triplet-problem 9

A Pythagorean triplet is a set of three natural numbers, $a < b < c$, for which,

$$a^2 + b^2 = c^2$$

For example, $3^2 + 4^2 = 9 + 16 = 25 = 5^2$.

There exists exactly one Pythagorean triplet for which $a + b + c = 1000$. Find the product abc .

ピタゴレアン

$$a^2 + b^2 = c^2$$

$$3^2 + 4^2 = 9 + 16 = 25 = 5^2$$

$a + b + c = 1000$ となるピタゴラス数の掛け算 ($a \times b \times c$) は

数値計算に誤差が出る

```
+ / 83.3333* 3 4 5  
1000
```

```
83.3333* 3 4 5  
250 333.333 416.666
```

```
30j10 ": */ 83.3333* 3 4 5x  
34722180.5555722120
```

```
+ / 1000* 3 4 5 * 1r12  
1000
```

```
20j4 ": */ 1000* 3 4 5 * 1r12  
34722222.2222
```

何れも正解とならない

```
 %: 200 300 500  
14.1421 17.3205 22.3607  
*/ %: 200 300 500  
5477.23
```

5477 は正解でない

JAPLA のワークショップで次の証明を教えてもらった

$$\begin{cases} a = m^2 - n^2 \\ b = 2mn \\ c = m^2 + n^2 \end{cases}$$

m,n をループで回す方法


```

pyt_sub1=: 3 : 0
NB. Usage: pyt_sub1 2-->2 1 3 4 5 12
ANS0=. <'
nr=. }.i. y
for_ctr. i. # nr do.
tmp0=. pyt_sub0 tmp1=. y, ctr{nr
  if. 1 = cp tmp0 do.
    ANS0=. ANS0, <tmp1,tmp0 , +/- tmp0
  end.
end.
}. >ANS0
)

```

```

pyt_sub0=: 3 : 0
' m0 n0'=: y
a=. -/ *: m0,n0
b=. +: */ m0,n0
c=. +/ *: m0,n0
a,b,c
)

```

```

cp=: check_phytagoean=: 3 : '(+/ *: 2{. y )= *: {: y'
NB. Usage: cp a b c
NB. cp 3 4 5 --> 1 // no is 0

```

方程式を解く方法もある。ループでも解はすぐに求まる。

3:1,2

4:1,2,3

と風潰しにピタゴリアンをもとめ、行きつ戻りつしながら合計が 1000 丁度になるものを探す。

10 Problem 10

The sum of the primes below 10 is $2 + 3 + 5 + 7 = 17$.

Find the sum of all the primes below two million.

10 に満たないの素数の合計は 17

200 万に満たない素数の合計は

10{. |. p: i.148934

2000003 1999993 1999979 1999969 1999957 1999891 1999889 1999871 1999861

この問いの Forum で Roger Hui が J で解の Script を書いていた