

# J でエクササイズ (その 1)

## 素数のドラゴン

SHIMURA Masato  
jcd02773@nifty.ne.jp

2012 年 1 月 29 日

### 目次

1	素数のドラゴン	1
1.1	J と素数	1
1.2	経過と解説	3
2	Script	7

### 概要

素数と素数の間にある数を並べてドラゴンを描く

## 1 素数のドラゴン

アメリカンサイエンスに永く連載された「マーチン・ガードナーの数学ゲーム」のうち初期に連載された赤  
 撮也・冬子訳の 22 編が「II(新装版)」として 2011 年 12 月に日経サイエンス社から出版された。いずれも水準  
 の高い力作であるので、幾つかを配列演算言語 J で料理してみよう。

これは第 4 章に載っている素数のドラゴンである

### 1.1 J と素数

J の素数生成の原始関数  $p$ : で求められる

`p: i.10 NB. 10 個生成`

`2 3 5 7 11 13 17 19 23 29`

素数の間隔 .

- 3 より大きいどのような素数も 6 の倍数  $\pm 1$  である.

J の剰余を求める関数 (Residue |) で確認する

`10 20 $ 6 | 2}. p:i.200`

`5 1 5 1 5 1 5 5 1 1 5 1 5 5 5 1 1 5 1 1`



```

+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| 5 | 5 | 5 | 1 | 13 | 1 | 13 | 9 | 3 | 7 | 5 | 5 | 3 | 9 | 9 | 5 | 5 | 3 | 3 | 7 | 7 | 5 | 1 |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|251|263|271|281|293|311|317|337|349|359|373|383|397|409|421|433|443|457|463|479|491|503|521|
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|257|269|277|283|307|313|331|347|353|367|379|389|401|419|431|439|449|461|467|487|499|509|523|
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| 5 | 1 | 3 | 9 | 3 | 3 | 5 | 1 | 5 | 5 | 3 | 7 | 7 | 1 | 1 | 3 | 7 | 1 | 11 | 3 | 3 | 11 | 17 |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

## 1.2 経過と解説

```
dragon0=: 3 : 0
```

```
NB. prime numbers
```

```
NB. Usage: dragon0 30
```

```
ind=:check_cutindex (tmp1=.: i. {: prime) e. prime =. p: i.y
```

```
tmp2=.,.>_2<\, . 2}. ind < ;.1 tmp1
```

```
tmp3=.(|. L:0 {"1 tmp2),. {"1 tmp2
```

```
('';1;2;''), konekone_sub tmp3
```

```
)
```

```
check_cutindex=: 3 : ' if. 0= { . y do. y=. 1,}.y end.'
```

```
NB. add 1 to cut-index, if top is 0
```

```
konekone_sub=: 3 : '(({: L:0{"1 y),.{: L:0{"1 y),.({. L:0{"1 y),.}.L:0 {"1 y'
```

```
prime p: 素数を求める
```

```
p: i.10
```

```
2 3 5 7 11 13 17 19 23 29
```

含まれるか否かの判別 members-of(*e.*) で判別する。

1 から素数の最後の数{: prime までの数列と素数を比較し、素数に 1、素数でなければ 0 の指標を作る

比較は *equal* = でもできるが *e.* の場合は 1 対 1 でなくとも含まれていればよく柔軟で使いやすい

```
|: (>:i.{:prime),: (>:i.{: prime) e. prime
```

```

1 0          11 1          21 0
2 1          12 0          22 0
3 1          13 1          23 1
4 0          14 0          24 0
5 1          15 0          25 0
6 0          16 0          26 0
7 1          17 1          27 0
8 0          18 0          28 0
9 0          19 1          29 1
10 0         20 0

```

カット 指標を用いて素数でない数を素数と関連ずけて分類する。指標によりカットする。

- カットの指標は最初は1。1でない場合は手に入れ替えか次を用いる  
`check_cutindex=: 3 : ' if. 0= {. y do. y=. 1,}.y end.'`
- `cut <;.1` で指標に従いリストを分割する

```

2 5 $ ind<;.1 tmp1
+-----+-----+-----+-----+-----+
|1   |2       |3 4 |5 6       |7 8 9 10   |
+-----+-----+-----+-----+
|11 12|13 14 15 16|17 18|19 20 21 22|23 24 25 26 27 28|
+-----+-----+-----+-----+

```

素数と素数以外の振分け 素数を抜く (各ボックス内の先頭を落とす } . L:0)

整列 取り、落とし、並べるの動詞

- `konekone_sub=: 3 : '({: L:0{"1 y),.{: L:0{"1 y),.({. L:0{"1 y),.}.L:0 {"1 y'`
- 

<i>take</i>	単項	{.	先頭	{:	末尾
	両項	n {.	前から	_n {.	後ろから
<i>drop</i>	単項	}.	先頭	}:	末尾
	両項	n }.	前から	_n }.	後ろから
<i>from</i>	両項	$n_0 n_1$ {	前から	$_n_0 _n_1$ {	後ろから

- `stich ,.` アレーを縦にして連結する。マトリクスやボックス横(サイド)に連結する

```

1 2 3, .4 5 6          (i.3 2), . 10 20 30
1 4                      0 1 10
2 5                      2 3 20
3 6                      4 5 30

```

- `rank "` アレーやマトリクス、ボックスへ作用させる方向を指定(変更)する

```

i. 3 4
0 1 2 3
4 5 6 7
8 9 10 11

```

```
{. i. 3 4      NB. デフォルト ランク 0
0 1 2 3
```

```
{. "1 i. 3 4   NB. 縦に作用させる ランクを 1 に変更
0 4 8
```

```
dragon=: 3 : 0
```

```
NB. number of prime numbers
```

```
NB. Usage: dragon 30
```

```
tmp1=. # L:0 tmp0=. dragon 0 y
```

```
({"1 tmp1),. (1 2 {"1 tmp0),. {"1 tmp1
)
```

```
number # 各ボックスの数を数える
```

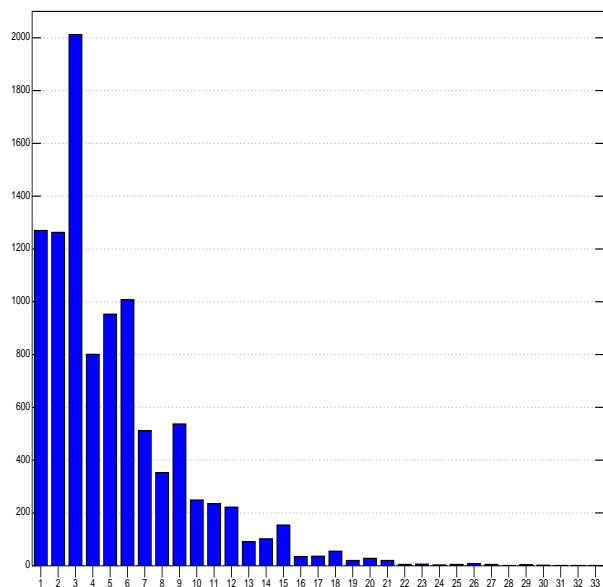
```
整理 素数以外の部分は数 (n) を表示
```

```
数のカウント . 素数は緻密かカウントしてみよう
```

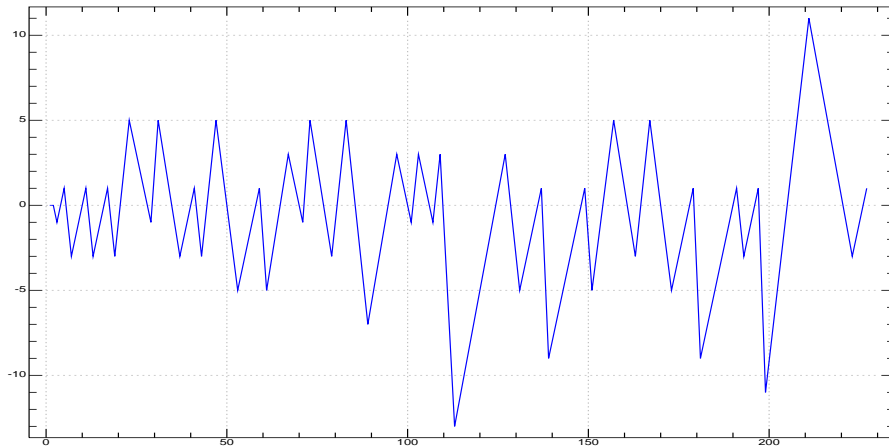
```
|: calc_dragon 1000
```

```
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31 33 NB. 素数と素数の間隔
174 169 244 83 100 75 42 26 34 10 12 12 3 4 8 1 1 NB. 間隔の分布
```

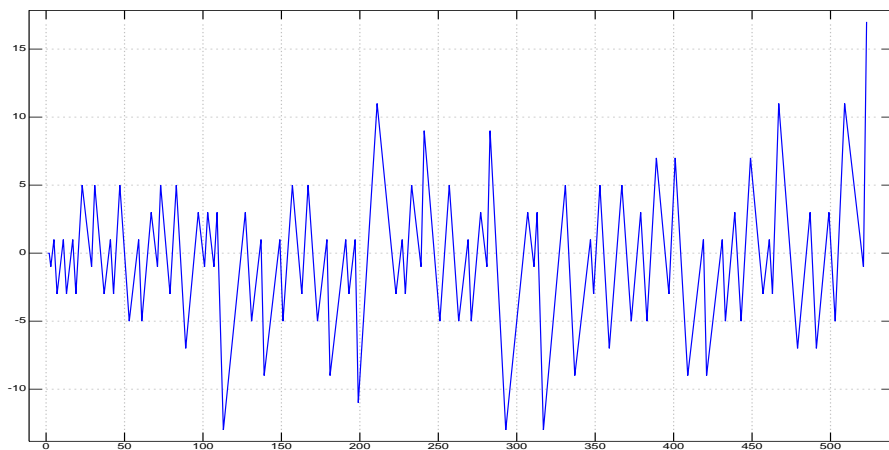
素数 10000 個とその間の数の分布



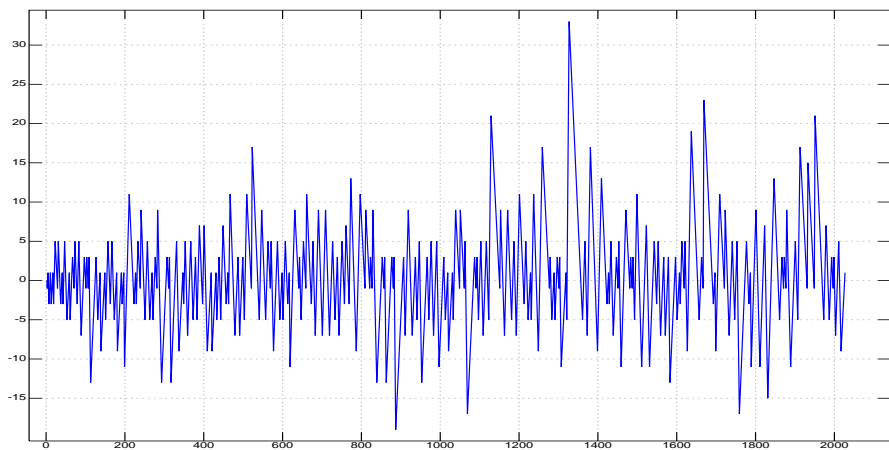
plot 本来は素数を一直線に右上がりに並べて素数で無い数をくねくねと手書きで付加するべきだが、簡略化し plot で片側をマイナスにして表す



dragon 50



dragon 100



dragon 2012

## 2 Script

```

NB. M.Gardner Math Games II Nr. 4
NB. Dragon or Snake
dragon0=: 3 : 0
NB. prime numbers
NB. Usage: dragon0 30
ind=:check_cutindex (tmp1=.: i. {: prime) e. prime =. p: i.y
tmp2=.,.>_2<\,., 2}. ind <;.1 tmp1
tmp3=.(|. L:0 {"1 tmp2),. {"1 tmp2
('';1;2;''), konekone_sub tmp3
)

check_cutindex=: 3 : ' if. 0= {. y do. y=. 1,}.y end.'
NB. add 1 to cut-index, if top is 0
konekone_sub=: 3 : '({: L:0{"1 y),.{: L:0{"1 y),.({. L:0{"1 y),.}.L:0 {"1 y'
NB. Konekone Mochi special

dragon=: 3 : 0
NB. count each not-prime numbers
NB. Usage: dragon 30
tmp1=. # L:0 tmp0=. dragon0 y          NB. count number
({."1 tmp1),. (1 2 {"1 tmp0),.{"1 tmp1 NB. n, prime ,prime ,n
)

calc_dragon=: 3 : 0
tmp0=.}. dragon y
tmp1=. /:~ |."1 ;("1) ((1 0){"1 tmp0), 2 3 {"1 tmp0
ind=.~: {. |: tmp1
(;{. L:0 tmp2),. ;# L:0 tmp2=. ind<;.1 {"1 tmp1
)

plot_dragon=: 3 : 0
NB. plot_dragon 308// nearly 2012
NB. make half of it to minus
tmp0=. dragon y
tmp1=. (- L:0 {"1 tmp0),. }. "1 tmp0
tmp2=. { |: /:~ ;"1 (1 0 {"1 tmp1),2 3{"1 tmp1
plot tmp2

```

)

## References

マーチン・ガードナー/赤堀也・冬子訳「マーチン・ガードナーの数学ゲーム II(新装版)」日経サイエンス社  
2011

J 言語の入手

<http://www.jsoftware.com> から DL する。

Script の入手 <http://japla.sakura.ne.jp>

Workshop 01/2012 から DL する