

Coin の並べ替えゲーム (Games of Coin Rearrangement)

(株)竹内ハガネ商行 システム室長 竹内寿一郎

1. はじめに

白と黒のコインが n 個ずつ横に並んでいる。このコインをすぐ隣にある2個のコインを同時に移動させて、 $2n$ 個のコインが1つおきに交互に並ぶように配置換えをしたい。ただし、コインは必ずすぐ隣に並んでいる2個を同時に動かし、また2個のコインの並び順は変えてはいけないものとする。また出来上がった交互に並んだ列は隙間があってはいけない。

ネットで調べてみると、 $2n$ 個のコインが交互に並んでいてそれを隣の2個を同時に移動させて n 個ずつの同色に並べ替えるというゲームがあり、本質的に同じゲームである。しかし、ネットで見る限り $n=3$ と $n=4$ の場合しか見られなかった。そして任意の n に対する統一的な解は得られないというようなことが書かれていたと思う。

このようなルールで n 個のコインの場合、最小何回で達成できるであろうか。今から30~40年前、何日か徹夜して、13か14個まで完成し、記録しておいた筈なのであるがその資料が散逸してしまったのである。しかし、うる覚えでどうにか $n=7$ まで再現してみたのであるが、歳のせいもありそれ以上はなかなか思い出せず、このたびJの力を借りて再現しようと思い立ったものである。

その時の経験によれば n 個のコインの場合、 $n=3$ 以上では、 n 回で出来ることが確かめられたのであるが、それが本当に最少の回数であるか、また、そのやり方は一意であるか、さらに n が13までは試みたが、 $n=14$ では2通りの答えが見つかったような気がしたりして、それから先はどうなるかなど、まだまだいろいろなことが分かっていない。

このゲームの肝いところは、ゲームの途中で前の段階に戻したいとき、前がどのような状態であったか全く分からなくなってしまうので、初めからやり直してみたりするのであるが、それもその段階まで行くのにもどのようにしたか忘れてしまうのである。このイライラする気持ちは大きな n に対して実行してみた人しか分からないであろう。

そこでこの冊子では解答を見つけるための援けとなる工夫をJで実現してみることにした。

2. $n=3$ の場合

$n=2$ の場合、どうやってもうまくゆかず、 $n=3$ 以上で考えることにする。

それでも $n=3$ は特別な場合で完成図も最初の配置から4個右へずれた配置図となってしまった。 $n=3$ の場合は3回で完成する。

0	○○○●●●××	順1、6、3、
1	○●●●○○××	
2	○●●××○●○	不動点 前5
3	●○○●○○	不動点 後5

3. $n=4$ の場合

これ以降、完成図を見てもらえばわかるように、右へ2個ずれて4回の試行で完成する。また

始めは必ず2番目3番から移動し、最後は1番目2番目を後ろから2番目3番目に挿入して完成する。この手順は裏返すと始めと終わりの手順が対称的になっていることに気が付く。N=3だけはこのルールに外れているがn=4からn=7までは当てはまっているし、以前私が試みたことを思い返してもn=14まで共通していえるルールである。

0行目の終わりに分かりやすく順として移動するコインの始めの位置を記しておいた。また、(n-1)とn行目の終わりに全く動かさなかったコインの位置も記しておいた。これらの情報はさらに大きなnについて試みるとき参考になるであろう。

0	○ <u>○○○</u> ●●●●●××	順2、5、8、1
1	○××○●●●●○○	
2	○●●○××●●○○	
3	○●●○●○○●××○	不動点 前4、7
4	××●○●○●○●○	不動点 後4、7

4. n=5の場合

0	○ <u>○○○</u> ○●●●●●××	順2、8、5、10、1
1	○××○○●●●●○○	
2	○●●○ <u>○○</u> ●●××●○○	
3	○●●○××○●○●●○○	
4	○●●○●○○●○○●××○	不動点 前4、7
5	××●○●○●○●○●○	不動点 後6、9

5. n=6の場合

0	○ <u>○○○</u> ○○●●●●●●××	順2、8、4、9、12、1
1	○××○○○●●●●●○○	
2	○●●○ <u>○○</u> ○●××●●●○○	
3	○●●××○●○○●●●○○	
4	○●●○●○○●○○××●●○○	
5	○●●○●○○●○○●○○××○	不動点 前6、7、11
6	××●○●○●○●○●○●○	不動点 後4、8、9

6. n=7の場合

0	○ <u>○○○</u> ○○○●●●●●●××	順2、11、5、10、7、14、1
1	○××○○○○●●●●●○○	
2	○●●○ <u>○○</u> ○●●●××●●○○	
3	○●●○××○●●●○○●●○○	
4	○●●○●○○●○○××○●●○○	
5	○●●○●○○××○●○○●●○○	
6	○●●○●○○●○○●○○●○○××○	不動点 前4、9、13

7 ××●○●○●○●○●○●○●○●○

不動点 後4、8、11

7. Jのプログラムと実行

F=:4 : 0

```
'i1 i2'=(i,>:i=.{.y)]['j1 j2'=(j,>:j=.{:y)
in=(i1,i2){x
a=((0,0),in)((i1,i2),j1,j2)}d=:c=:x
)
```

NB. Init 4

Init=:3 : 0

```
a=(N#1),(N#2),0,k=:0[A=:i.0 0[E=: 'x','o',(7{a.)][j=:2*N=:y
A=:A,a[T=:1 2$0[a2=:((1 2),_4).2}.a0=:0 0,(2*N)$ (2 1)),2 9 9 1
)
```

NB. 4: 2 5 8 1(1 4 7 0)

NB. CoinGame 4

CoinGame=:3 : 0

Init y

label_Loop.

z=:input (1)

i=: ".z

print i,j

if. z='e' do. goto_OWARI. end.

if. z='b' do.

z=:input 1[print 'Bach_Numbers'

if. */z e.'0123456789' do.

print 'Backward'

j=:{:T

T=:}:T

A=:}:A

a=:{:A

print (":k=:<k), ' ',":{:T

print a

end.

else.

print 'Forward'

(k=:k+1){T=:T,(i,j)

a F i,j

```
j:=i[A=:A,a
end.
print 'k=',":k[print a
print 'ItchiDo=',":(+/a=a2),+/a=a0
if. a-:a0 do. goto_OWARI. end.
goto_Loop.
label_OWARI.
print 'END'
)
CoinGame 5
1
1 10
Forward
1 0 0 1 1 2 2 2 2 1 1
k=1
ItchiDo=5 5
7
7 1
Forward
1 2 2 1 1 2 2 0 0 2 1 1
k=2
ItchiDo=6 4
4
4 7
Forward
1 2 2 1 0 0 2 1 2 2 1 1
k=3
ItchiDo=8 6
9
9 4
Forward
1 2 2 1 2 1 2 1 2 0 0 1
k=4
ItchiDo=10 8
0
0 9
Forward
0 0 2 1 2 1 2 1 2 1 2 1
```

k=5
ItchiDo=8 12
END
END
a{E
XX00000
A{E
00000XX
0XX0000
000XX00
00XX000
0000XX0
XX00000

8. 参考文献

n=3の場合

http://www.imd-g.com/puz01_2_8.htm

<http://r27.jp/quiz/tait-coin-odd/>

n=4の場合

http://www.imd-g.com/puz01_2_18.htm

<http://ameblo.jp/masanori432/entry-10074234174.html>

<http://r27.jp/quiz/tait-coin/>