黑白棋子问题模型

一、问题重述

有黑色与白色棋子共n个围成一圈，现相其中插入新棋子，插入规则是：若原本的棋子与其下一颗棋子颜色相同，则在它们中间插入黑色棋子，反之则插入白色棋子。如此进行完一圈插入操作后，撤掉原本的棋子。随着这样的操作一圈一圈进行，棋子的颜色将会如何变化?

二、问题分析

由于题目中操作“若原本的棋子与其下一颗棋子颜色相同，则在它们中间插入黑色棋子，反之则插入白色棋子”，我们观察到操作具有“将原本的两种属性利用其相互之间的关系再次变换为两种属性之一”这样的特征。接下来，为完成模型的构建，我们需要在数学领域中找到符合这个特征的操作以及对象。

三、模型的假设

在查询过后，我们发现”-1“与”1“的乘法运算恰巧满足该规律，具体解释如下：

- 首先我们设置黑棋为”+1“，白棋为”-1“，
- 当我们模拟插入新棋子的操作时，对数字进行乘法运算，若原本两数不同，则运算结果为”-1“，恰好代表白棋，而在两数都为”+1“情况下，运算结果为”+1“，恰好代表黑棋，与题目中对于棋子的操作逻辑吻合

四、模型的建立

由模型的假设（三）部分我们可以知道，下一轮 a_i 处棋子的颜色可以用原本 a_i*a_{i+1} 的结果来表示，那么接下来我们就可以实现逐圈插入棋子的操作。

任取一个位置的棋子，记其位置为1，则顺时针旋转，棋子的位置依次是2，3，4......

这里我们首先以三个棋子举例，将三个棋子的颜色分别表示为a_1、a_2、a_3，其每轮状态变换如下：

> $a_1 \to a_1a_2 \to a_1a_2^2a_3 \to a_1^2a_2^3a_3^3 \to$ ......
>
> $a_2 \to a_2a_3 \to a_2a_3^2a_1 \to a_2^2a_3^3a_1^3 \to$ ......
>
> $a_3 \to a_3a_1 \to a_3a_1^2a_2 \to a_3^2a_1^3a_2^3 \to$ ......(进行三轮操作)

其变换逻辑即如上图，接下来我们尝试四个棋子，同样设置变量：a_1、a_2、a_3、a_4, 其变换如下：

> $a_1 \to a_1a_2 \to a_1a_2^2a_3 \to a_1a_2^3a_3^3a_4 \to a_1^2a_2^4a_3^6a_4^4 \to$ ......
>
> $a_2 \to a_2a_3 \to a_2a_3^2a_4 \to a_2a_3^3a_4^3a_1 \to a_2^2a_3^4a_4^6a_1^4 \to$ ......
>
> $a_3 \to a_3a_4 \to a_3a_4^2a_1 \to a_3a_4^3a_1^3a_2 \to a_3^2a_4^4a_1^6a_2^4 \to$ ......
>
> $a_4 \to a_4a_1 \to a_4a_1^2a_2 \to a_4a_1^3a_2^3a_3 \to a_4^2a_1^4a_2^6a_3^4 \to$ ......（进行四轮操作）
>
> (进行完四轮操作后，a_i的指数都是偶数，证明其背后的数字一定为+1，也就是都变成了黑棋，此后不再变化)

由此可以扩展到 n 轮操作;

> $a_1 \to a_1a_2 \to a_1a_2^2a_3 \to a_1a_2^3a_3^3a_4 \to a_1a_2^4a_3^6a_4^4a_5 \to$ ......

> $a_2 \to a_2a_3 \to a_2{a_3}^2a_4 \to a_2{a_3}^3{a_4}^3a_5 \to a_2{a_3}^4{a_4}^6{a_5}^4a_6 \to$ ......
>
> $a_3 \to a_3a_4 \to a_3{a_4}^2a_5 \to a_3{a_4}^3{a_5}^3a_6 \to a_3{a_4}^4{a_5}^6{a_6}^4a_7 \to$ ......
>
> ......
>
> ......
>
> $a_n \to a_na_1 \to a_n{a_1}^2a_2 \to a_n{a_1}^3{a_2}^3a_3 \to a_n{a_1}^4{a_2}^6{a_3}^4a_4 \to$ ......

即对于该模型，我们对于结果的判断取决于操作进行到第 k 轮后 a_i的指数，若所有 a_i 的指数都为偶数，那么其乘积必定为1，则代表都换成了黑棋，否则还存在其他可能。

五、模型的求解

1. 实际上，在从模型的建立（四）中少数几个棋子的操作中不难发现，设棋子总数为 n ， 当对位置为 i 的棋子进行 k (k<=n) 轮操作后，其结果状态是一个从 a_i 开始的乘法序列，而可以观察到从a_i开始，各因数的指数恰好构成了杨辉三角的第 k+1 行序列,且当k等于棋子的总数n时，该序列末尾位置的1会合并到首位置的1上变为2。例如上述对四个棋子的操作中，进行第三轮操作后，棋子a_i的状态$a_1{a_2}^3{a_3}^3a_4$各因数的指数正是杨辉三角的第四行序列。其他位置同理，其证明如下：

证明：

对于n个棋子，在经历一轮变换后，a_1 变为 a_1a_2 ， a_2 变为 a_2a_3 ,不难发现两个位置结果的表示上，指数序列没有发生变化，只是对应棋子的编号发生了改动，且该改动是平移向后挪动的，即对于这里的第一轮操作结果，指数的序列都为[1,1],而位置1棋子下标是[1,2],位置1棋子下标是[2,3]，而每一次的结果都由递推得到，我们可以利用数学归纳法证明对于每一轮变换后的结果都有该结论。

这里我们讨论 a_i,若a_i经过k轮后，其结果为 $b_i=a_ia_{i+1}{}^ka_{i+2}{}^la_{i+3}{}^ma_{i+4}{}^n$...... 那么此时$a_{i+1}$的结果应为 $b_{i+1}=a_{i+1}a_{i+2}{}^ka_{i+3}{}^la_{i+4}{}^ma_{i+5}{}^n$......,那么在进行一轮操作，$a_i$ 的结果变为 b_ib_{i+1}，实际上将对应位置的指数相加即可，也就是 a_{i+1} 的指数会变为原本 a_{i+1} 的指数加上 a_{i+1} 的指数，即 $b'=a_ia_{i+1}{}^{k+1}a_{i+2}{}^{l+k}a_{i+3}{}^{m+l}a_{i+4}{}^{n+m}$......,这种变化恰好与杨辉三角的运算逻辑相同。

最后我们讨论杨辉三角中的"1"，实际上它对应的是经过k轮变换后a_i的指数，经过一轮操作后a_i状态传递到 i-1 位置的棋子上，再过一轮又传递到 i-2 位置棋子上，以此类推，可以知道当经过 n 轮操作后，a_i 的状态才第一次 反馈到 i 位置自身的结果，此时结果中 a_i 的指数变为2。

得证。

2. 接下来我们便需要讨论杨辉三角第k行序列的性质，这里我们查阅资料发现对于杨辉三角的第 2^k+1 行，其除了首位和末尾的1，其他数字均为偶数（棋子的颜色变化问题[1]),在这种情况下，再加上刚才1中提到序列中首末位1的合并，我们很容易得到对于 $n=2^k$ 个棋子，若对其进行至多 n 轮操作，其结果的表示必定都为+1，也就是都变成了黑棋。
3. 而对于 $n\neq2^k$时，我们并未观察到杨辉三角第 n 行以及附近甚至其他行有什么特殊且相关的性质，我们认为，这种情况下，棋子的变化不具备特殊规律，较大可能黑白棋子会一直存在，只有较小的可能会全部变为黑子

-结论：如果棋子总数 $n=2^k$,则至多经过 n 次操作，棋子会全部变为黑色，若不满足该条件，棋子的变化并没有明显规律

六、模型的检验

下面我们用matlab来进行模拟：

- 首先我们编写了一个matlab脚本程序，该程序要求输入棋子的总数以及需要操作的轮数，随机初始化棋子的颜色分布，如何输出每一轮操作后的棋子状态。

1. 针对 n=2^k的情况

   - 对于 n=8,有如下结果：

```
>> demo1
棋子数：8
迭代次数：8
```

```
命令行窗口
x0 =
     1    -1     1    -1    -1     1    -1     1

i =
     1

x1 =
    -1    -1    -1     1    -1    -1    -1     1

i =
     2

x1 =
     1     1    -1    -1     1     1    -1    -1

i =
     3

x1 =
     1    -1     1    -1     1    -1     1    -1

i =
     4

x1 =
    -1    -1    -1    -1    -1    -1    -1    -1

i =
     5

x1 =
     1     1     1     1     1     1     1     1

i =
     6

x1 =
     1     1     1     1     1     1     1     1
```

```
i =
     7

x1 =
     1     1     1     1     1     1     1     1

i =
     8

x1 =
     1     1     1     1     1     1     1     1

>>
```

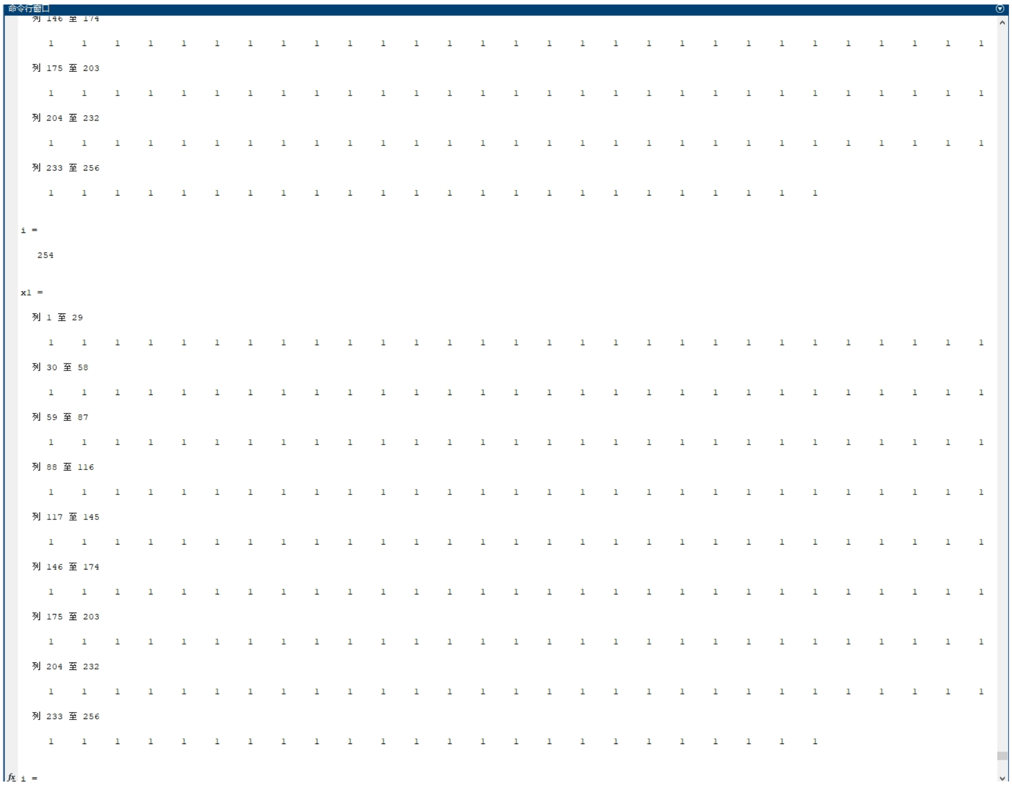
   - 对于 n=64 的情况，如下：

```
命令行窗口
    -1     1    -1     1    -1     1

i =

    60

x1 =

  列 1 至 29

    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1

  列 30 至 58

    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1

  列 59 至 64

    -1    -1    -1    -1    -1    -1

i =

    61

x1 =

  列 1 至 29

     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1

  列 30 至 58

     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1

  列 59 至 64

     1     1     1     1     1     1

i =

    62

x1 =

  列 1 至 29

     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1

  列 30 至 58

     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1

  列 59 至 64
```

可以看到在进行到第61轮操作后，棋子已经都变成了黑色

- 最后我们再来测试 n=256 的情况：

```
  列 88 至 116

    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1

  列 117 至 145

    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1

  列 146 至 174

    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1

  列 175 至 203

    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1

  列 204 至 232

    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1

  列 233 至 256

    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1    -1

i =

   253

x1 =

  列 1 至 29

     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1

  列 30 至 58

     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1

  列 59 至 87

     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1

  列 88 至 116

     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1

  列 117 至 145

     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1

  列 146 至 174

     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1

  列 175 至 203

     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1     1
```


```
命令行窗口
  列 146 至 174
    1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  列 175 至 203
    1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  列 204 至 232
    1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  列 233 至 256
    1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

i =

  254

x1 =

  列 1 至 29
    1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  列 30 至 58
    1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  列 59 至 87
    1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  列 88 至 116
    1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  列 117 至 145
    1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  列 146 至 174
    1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  列 175 至 203
    1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  列 204 至 232
    1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
  列 233 至 256
    1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

fx i =
```

可以看到在进行第253次操作后棋子已经都变成黑色

由此可以证明对于 $n=2^k$ 情况下，最多对棋子进行 n 轮操作，便可实现全为黑棋

2. 对于 $n\neq 2^k$情况：

   - 首先看 n=3 时：

```
棋子数: 3
迭代次数: 100

k =

    0.7624


k =

    0.0073


k =

    0.6800


x0 =

     1    -1     1
```

```
i =

    95


x1 =

     1    -1    -1


i =

    96


x1 =

    -1     1    -1


i =

    97


x1 =

    -1    -1     1


i =

    98


x1 =

     1    -1    -1


i =

    99


x1 =

    -1     1    -1


i =

   100


x1 =

    -1    -1     1
```

我们发现在[1,-1,1]这样的分布下,即便执行100次操作也无法变为全为黑棋/白棋

之后我们对 n=5~20且$n \neq 2^k$,n==57,n=255 等进行了多轮验证，设定操作轮数为1000，结果依然没有得到全黑/全白的结果，基本符合我们的判断

- **由此得到，如果棋子总数 $n=2^k$,则至多经过 n 次操作，棋子会全部变为黑色，若不满足该条件，棋子的变化并没有明显规律。**

七、模型评价

该模型主要针对 $n=2^k$情况给出证明，当 $n=\neq 2^k$时，并未找到规律

参考文献

[1]窦霁虹,郭明焕,崔志明.棋子的颜色变化问题[J].西北大学学报(自然科学版),2000,(06):470-472.

附录

matlab脚本程序:

```
n = input('请输入棋子数：');
times = input('请输入迭代次数：');
x0 = zeros(1,n);
x1 = zeros(1,n);
for i=1:n
    k = rand(1,1)
    if (k>0.5)
        x0(i) = 1;
    else
        x0(i) = -1;
    end
end
x0
for i=1:times
    i
    for k = 1:n-1
        x1(k)=x0(k)*x0(k+1);
    end
    x1(n)=x0(n)*x0(1);
    x1
    x0=x1;
end
```