Shenyunhan and His Tiling Job
题解

Lancern

2019 年 12 月 12 日

本题定位为一道中等难度的思维题、找规律题。题目大意为，给定一个由正六边形密接而拼成的倒三角形图案（如右图所示），已知第一行（倒三角形底边）的染色方案，求按照如下的规则，使用红、绿、蓝三种颜色对余下的正六边形进行染色后，位于倒三角形顶点位置处的正六边形的颜色：

1. 若处在当前待染色的正六边形正上方的两个正六边形被染成不同的颜色，则当前的正六边形将被染成三种颜色中没有出现的那一种颜色；
2. 若处在当前待染色的正六边形正上方的两个正六边形被染成相同的颜色，则当前的正六边形将被染成同样的颜色。

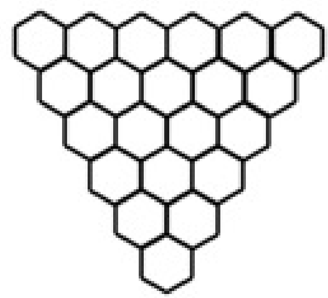

图 1: 正六边形密接模式

本题解分为两部分：第一部分阐述该染色方案导出的规律，第二部分阐述代码实现。

1 规律

定义 1. 记第一行具有 n 个正六边形的拼接图案为 n-拼接。

由定义1可知，图1中展示的拼接方案为一个 6-拼接。

定理 1. 当 $n \geq 2$ 时，若有 $n = 3^k + 1$，其中 k 为一非负整数，则在 n-拼接的任意一合法染色方案下，位于顶角处的正六边形的颜色仅取决于第一行中处于最左和处于最右的正六边形的颜色，且顶角处的正六边形颜色可以按照原染色规则从这两个正六边形导出。

证明．定理1可由数学归纳法证明。

当 $n=2$ 时，原命题平凡，在此略过；

当 $n=4$ 时，原命题有两种证明方法。考虑到此时所有合法的染色方案仅有 $3^4=81$ 种，因此可以使用穷举法进行证明，在此不做展开。下面重点阐述第二种证明方法。

首先给三种颜色进行编码：令 1 表示红色，2 表示绿色，3 表示蓝色。记 $x\oplus y$ 表示当位于当前正六边形正上方的两个正六边形的颜色分别为 x 和 y 时，当前正六边形应当染成的颜色。则有：

$$x\oplus y=(-x-y)\mod 3+3 \tag{1}$$

其中 mod 表示取余操作，且余数的符号与被除数符号一致。

设当 $n=4$ 时，位于第 i 行的从左向右第 j 个正六边形的颜色为 $c_{i,j}$。则有：

$$\begin{bmatrix} c_{1,1} & c_{1,2} & c_{1,3} & c_{1,4} \\ c_{2,1} & c_{2,2} & c_{2,3} & \\ c_{3,1} & c_{3,2} & & \\ c_{4,1} & & & \end{bmatrix}=\begin{bmatrix} a & b & c & d \\ -a-b & -b-c & -c-d & \\ a+2b+c & b+2c+d & & \\ -a-3b-3c-d & & & \end{bmatrix}\mod 3+3 \tag{2}$$

因此有：

$$\begin{aligned} c_{4,1}&=(-a-3b-3c-d)\mod 3+3 && (3)\\ &=(-a-d)\mod 3+3 && (4)\\ &=c_{1,1}\oplus c_{1,4} && (5) \end{aligned}$$

因此当 $n=4$ 时原命题得证。

当 $n>4$ 且 $n=3^k+1$ 时，假设原命题在 $n=3^{k-1}+1$ 下成立，即有：

$$c_{3^{k-1}+1,1}=c_{1,1}\oplus c_{1,3^{k-1}+1} \tag{6}$$

由 n-拼接的自相似性容易推广得到：

$$c_{i,j}=c_{i-3^{k-1},j}\oplus c_{i-3^{k-1},j+3^{k-1}},i\in[1,n],j\in[1,n-i+1] \tag{7}$$

注意到此时有：

$$\begin{bmatrix} c_{1,1} & c_{1,3^{k-1}+1} & c_{1,2\times 3^{k-1}+1} & c_{1,3^k+1} \\ c_{3^{k-1}+1,1} & c_{3^{k-1}+1,3^{k-1}+1} & c_{3^{k-1}+1,2\times 3^{k-1}+1} & \\ c_{2\times 3^{k-1}+1,1} & c_{2\times 3^{k-1}+1,3^{k-1}+1} & & \\ c_{3^k+1,1} & & & \end{bmatrix}$$

与

$$\begin{bmatrix} c_{1,1} & c_{1,2} & c_{1,3} & c_{1,4} \\ c_{2,1} & c_{2,2} & c_{2,3} & \\ c_{3,1} & c_{3,2} & & \\ c_{4,1} & & & \end{bmatrix}$$

在结构上相似，因此当 $n = 3^k + 1$ 时原命题成立。

综上所述，当 $n = 3^k + 1$ 时原命题成立。证毕。 □

2 实现

根据节1中发现的规律，可以得到如下的递归实现算法。该算法计算当位于第一行正六边形的颜色序列为 $\{a_i\}$ 时，位于第 row 行第 col 列正六边形的颜色：

function GetColorAt($\{a_i\}, row, col$)
  if $row = 1$ **then**
    return a_{col}
  else
    $e := \lfloor \log_3 (row - 1) \rfloor$
    $s := 3^e$
    $lhs :=$ GetColorAt($a, row - s, col$)
    $rhs :=$ GetColorAt($a, row - s, col + s$)
    return $lhs \oplus rhs$
  end if
end function

易知，本题求解的最终答案即为 GetColorAt$(n, 1)$。下面对该算法的时间复杂度进行分析。

定理 2. 算法 GetColorAt 的时间复杂度为 $O(\log n)$。

证明. 设 GetColorAt$(\{a_i\}, n, m)$ 的运行时间为 $T(n)$。则有：

$$T(n) = \begin{cases} 2T\left(n - 3^{\lfloor \log_3 (n-1) \rfloor}\right) + O(1) & \text{if } n > 1 \\ O(1) & \text{if } n = 1 \end{cases} \tag{8}$$

可以进一步化简为：

$$T(n) = \begin{cases} 2T\left(\frac{1}{3}n\right) + O(1) & \text{if } n > 1 \\ O(1) & \text{if } n = 1 \end{cases} \tag{9}$$

由主定理[1]可以直接推出：

$$T(n) = O(\log n) \tag{10}$$

命题得证。 □

[1] 主定理（Master Theorem）：https://en.wikipedia.org/wiki/Master_theorem_(analysis_of_algorithms)