

从特殊情况考虑

复旦附中 李天翼

引子

- 从特殊情况考虑是一种重要的数学思想。
- 在算法设计中，巧妙地运用这一思想，可以取得事半功倍的效果。

- 特殊情况主要分为两种：极端情况和简单情况

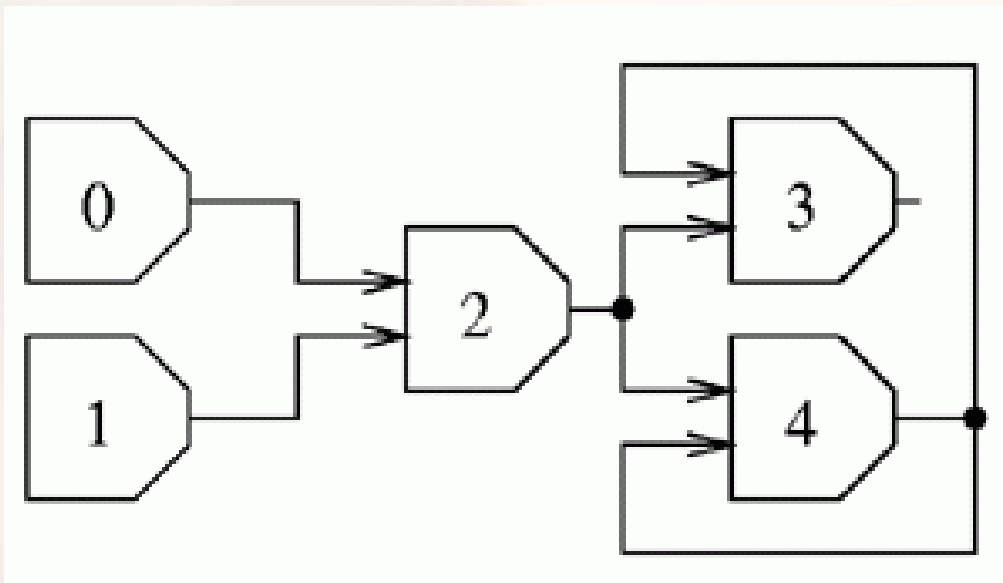
极端情况：例 1 Bra

简单情况：例 2 Sko

问题一 (POI 03-04 Bra 改编)

问题描述

- 如图，给定 n 个门，分别编号为 0 至 $n-1$ 。每个门可能有多个输入端，但只有一个输出端。
- 电路的信号有三种可能性： 0 、 $1/2$ 和 1 。



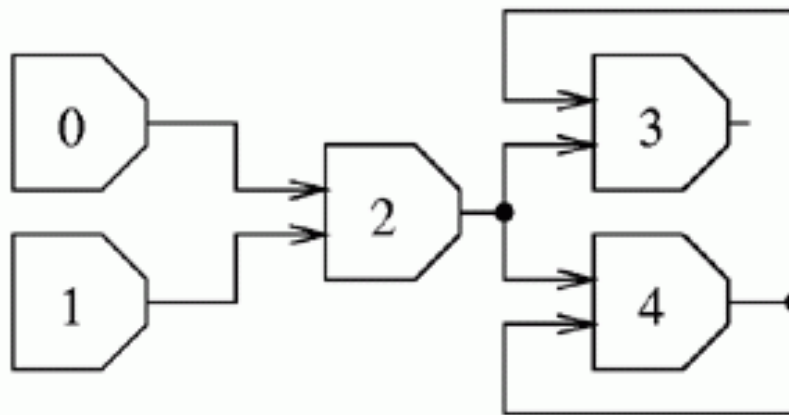
问题一

问题描述（续）

- 编号为 0 和 1 的门没有输入端，0 号门始终输出 0，1 号门始终输出 1。对于其它的门，它的输入信号中
- 0 的个数比 1 的个数多时，它输出 0；
- 1 的个数比 0 的个数多时，它输出 1；
- 0 的个数和 1 的个数一样多时，它输出 $1/2$ ；
- 保证存在符合要求的输出状态。
- 给定一个电路，要求尽可能多地确定每个门的输出结果。

问题一 初步思考

- 令 $C_{j,i}$ 表示 i 号门的所有输入端中，来自 j 号门输出端的数量。设 $P(i)$ 为 i 号门的输出状态。



- $C_{2,0}=1, C_{2,1}=1, C_{3,2}=1, C_{3,4}=1, C_{4,2}=1, C_{4,4}=1$
其它均为 0

问题一

初步思考（续）

- 令
$$U(i) = \frac{\sum_{j=0}^{n-1} C_{j,i} P(j)}{\sum_{j=0}^{n-1} C_{j,i}}$$

即相当于 $i(i>1)$ 号门所有输入信号的平均值。

- $U(i) < 1/2$, 则 $P(i) = 0$
- $U(i) = 1/2$, 则 $P(i) = 1/2$
- $U(i) > 1/2$, 则 $P(i) = 1$

问题一 考虑极端情况

- 令 $P_{\min}(i)$ 和 $P_{\max}(i)$ 分别为 $P(i)$ 在所有可能的电路中能取到的最小值和最大值，它们是 $P(i)$ 的**极端**情况。
- 显然，若 $P_{\min}(i)=P_{\max}(i)$ ($0 \leq i \leq n-1$)，则 i 号门的输出状态是固定的，否则就不是固定的。
- 因此，我们只需要求出 $P_{\min}(i)$ 和 $P_{\max}(i)$ 。

问题一 一个想法

- 如果 $P(k)$ 变大，那么所有的 $U(i)$ ($0 \leq i \leq n-1$) 也只能变大，而不可能变小，而对应的 $P(i)$ 也是如此。
- 令 $P(i) = P_{\min}(i)$ ($0 \leq i \leq n-1$)，可以得到，此时的输出状态是符合要求的。

问题一

$P_{\min}(i)$ 的求法

- 不妨先将所有的门的输出状态都标为 0，此时只有 1 号门不正确。从 1 号门开始，将它的输出状态改为 1。然后不断找到矛盾所在，进行迭代。由于每个门的状态最多变两次（0 变 1/2，1/2 变 1），每个门的输入的总数不超过 200000，因此在不超过 $2 \times 200000 = 400000$ 次迭代后，迭代就会中止。迭代终止时，有 $P(i) = P_{\min}(i)$ ($0 \leq i \leq n-1$)。
- 类似的，也可以求得 $P_{\max}(i)$ ($0 \leq i \leq n-1$)。

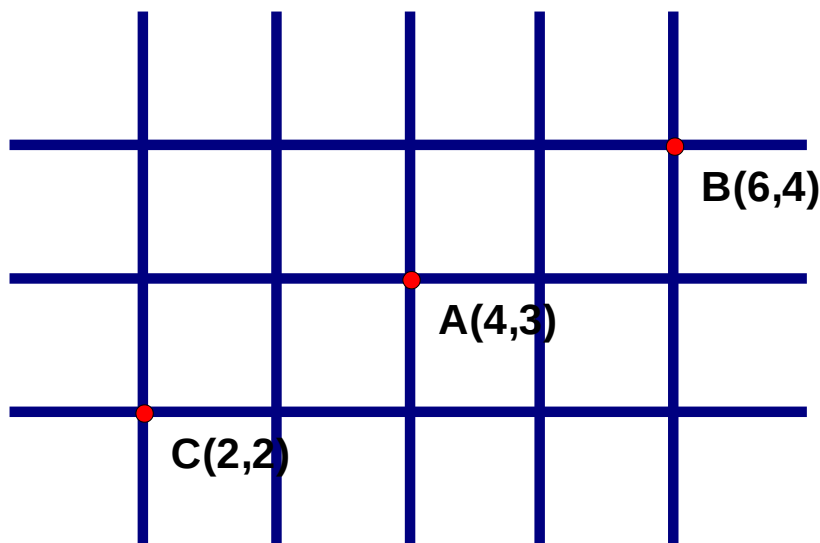
问题一 小结

- 极端情况是特殊情况的一种表现形式。题目中的许多性质，往往会通过一些具有极端性质的对象（比如本题中的极值）表现出来。这就使得我们可以以它们为重点考察对象，来寻找突破口和答案。



问题二 (POI 04-05 Sko 改编)

- 骑士在一个无限大的坐标平面上移动。他能够执行的每种移动可以用一对整数 (a,b) 来描述，这表示骑士可以从 (x,y) 移动到 $(x+a,y+b)$ 或 $(x-a,y-b)$



如果一个骑士有移动 $(2,1)$ ，那么他在 $A(4,3)$ 点就可以移动到 $B(6,4)$ 或 $C(2,2)$

问题二

题目描述（续）

- 对于每个骑士，从原点出发所能够到达的点，不全在一条直线上
- 给定一个骑士和一些点，问这个骑士从原点出发，能到达这些点中的哪几个。

为了讨论方便，设给定骑士的第 \mathbf{k} 种移动为 $(\mathbf{a_k}, \mathbf{b_k})$ 。将一个骑士从原点出发，能够到达的点称为该骑士的可行点。一个骑士的可行点的集合称为该骑士的可行点集。

问题二

考虑一维情况

- 棋盘是二维的，我们不妨先考虑比较**简单**的一维情况。
- 此时，每个骑士都在一条直线上移动，他的第 **i** 种移动 (**$1 \leq i \leq n$**) 可以表示为一个整数 (**a_i**)，令 **$r = (a_1, a_2, \dots, a_n)$** ，则他能够到达横坐标为 **$X$** 的点的充要条件是 **$r|X$** 。

问题二

回归二维情况

- 对于不退化的二维情况（即骑士所能够到达的点，不全在一条直线上），可以猜想他的可行点集与满足下列三个条件之一的所有点的集合相同。
- **(1) $r|X+Y$ (2) $r|X+qY$ (3) $r|pX+qY$ (p, q, r 均为待定整数, $r \neq 0$)**。
- 通过对**简单**情况的验证，易知前两种假设是错误的。对于最后一种假设，由于参数比较多，一时难以判断其是否正确。

问题二 猜想

- 设 $S(p,q,r)$ 为所有满足 $r|pX+qY$ (p,q,r 均为整数, $r \neq 0$) 的点的集合。
- 考虑猜想: 对于不退化的二维情况 (即骑士所能够到达的点, 不全在一条直线上), 存在整数 p,q,r , $r \neq 0$, 满足骑士的可行点集与 $S(p,q,r)$ 相同。

问题二

猜想的简化

- 由于 n 的上限是 **100**，比较大，可以先考虑比较**简单**的 $n=2$ ($n=2$ 是最小的非退化情况)。可以用数学归纳法证明，只需要讨论 $n=2$ 。
- 证明要点：

假设只有前 k 种移动的骑士的可行点集与 $S(p, q, r_0)$ 相同，设 $r = (pa_{k+1} + qb_{k+1}, r_0)$ 那么有全部 $k+1$ 种移动的骑士的可行点集与 $S(p, q, r)$ 相同。



问题二

对 $n > 2$ 的处理

- 假设当 $n=k$ (k 为任意正整数) 时猜想成立, 当 $n=k+1$ 时, 设只有前 k 种移动的骑士的可行点集与 $S(p,q,r_0)$ 相同。则对有全部 $k+1$ 种移动的骑士而言, (X,Y) 是他的可行点的充要条件是: 存在整数 m 满足 $p(X-ma_{k+1})+q(Y-mb_{k+1}) \equiv 0(\text{mod } r_0)$ 。
- 上式等价于 $(pa_{k+1}+qb_{k+1})m \equiv pX+qY(\text{mod } r_0)$ 。考虑这个模线性方程, 知它有解等价于 $(pa_{k+1}+qb_{k+1}, r_0) \mid pX+qY$ 。令 $r=(pa_{k+1}+qb_{k+1}, r_0)$, 知当 $n=k+1$ 时, 猜想也成立。

问题二 命题 1

- 当 $n=2$ 时，骑士只有两种移动 (a_1, b_1) 和 (a_2, b_2) 。且 $a_1 b_2 - a_2 b_1 \neq 0$ （否则退化为一维情况）。
- 为了能更好地挖掘这个问题的本质，不妨证明简单一点的命题 1。
- 命题 1: 骑士有两种移动 (a_1, b_1) 和 (a_2, b_2) ， $a_1 b_2 - a_2 b_1 \neq 0$ ， a_1 和 a_2 互质， b_1 和 b_2 互质，存在整数 p, q, r ， $r \neq 0$ ，满足骑士的可行点集与 $S(p, q, r)$ 相同。
- 命题 1 与原猜想的本质区别在于，它增加了一个条件： a_1 和 a_2 互质， b_1 和 b_2 互质。

问题二

命题 1 的构造性证明

- 命题 1 中， $\mathbf{p}, \mathbf{q}, \mathbf{r}$ 三个值均为待定，可变的数太多了，难以考虑。
- 根据裴蜀定理，存在整数 \mathbf{x}, \mathbf{y} 满足 $\mathbf{a}_1\mathbf{x} + \mathbf{a}_2\mathbf{y} = 1$ 设 $\mathbf{s}_1, \mathbf{s}_2$ 为满足 $\mathbf{a}_1\mathbf{s}_1 + \mathbf{a}_2\mathbf{s}_2 = 1$ 的两个整数。
- 令 $\mathbf{p} = \mathbf{b}_1\mathbf{s}_1 + \mathbf{b}_2\mathbf{s}_2, \mathbf{q} = -1, \mathbf{r} = \mathbf{a}_1\mathbf{b}_2 - \mathbf{a}_2\mathbf{b}_1$
- 能够证明，命题 1 成立。



问题二 命题 1 的证明——必要性

➤ 先证必要性，即若点 (X,Y) 是骑士的可行点，则 $r \mid pX + qY$ 。

考虑 $pa_1 + qb_1$

$$\begin{aligned} pa_1 + qb_1 &= (b_1s_1 + b_2s_2) a_1 - b_1 = (a_1s_1 - 1)b_1 + b_2s_2a_1 \\ &= -a_2s_2b_2 + a_1s_2b_2 = s_2(a_1b_2 - a_2b_1) = s_2r \end{aligned}$$

同理 $pa_2 + qb_2 = -s_1r$

由于点 (X,Y) 是骑士的可行点，设骑士将移动 (a_1, b_1) 执行 c 次，移动 (a_2, b_2) 执行 d 次后，从原点到达点 (X,Y) 。

问题二

命题 1 的证明——必要性

$$\begin{aligned} Q \quad pX + qY &= p(ca_1 + da_2) + q(cb_1 + db_2) \\ &= c(pa_1 + qb_1) + d(pa_2 + qb_2) = cs_2r - ds_1r = r(cs_2 - ds_1) \\ \therefore r &\mid pX + qY \end{aligned}$$

问题二

命题 1 的证明——充分性

➤再证充分性，即若 $r \mid pX + qY$ ，则点 (X, Y) 是骑士的可行点。

由于 $r \mid pX + qY$ ，设 $pX + qY = kr$ ，即 $pX - Y = kr$

问题二

命题 1 的证明——充分性

$$\begin{aligned} Q \quad X &= X \times 1 + 0 = X (a_1 s_1 + a_2 s_2) + (ka_1 a_2 - ka_1 a_2) \\ &= (Xa_1 s_1 + ka_1 a_2) + (Xa_2 s_2 - ka_1 a_2) \\ &= (Xs_1 + ka_2)a_1 + (Xs_2 - ka_1)a_2 \\ Y &= pX + (Y - pX) = Xp - kr \\ &= X (b_1 s_1 + b_2 s_2) b_1 - k (a_1 b_2 - a_2 b_1) \\ &= (Xb_1 s_1 + ka_2 b_1) + (Xb_2 s_2 - ka_1 b_2) \\ &= (Xs_1 + ka_2)b_1 + (Xs_2 - ka_1)b_2 \end{aligned}$$

\therefore 骑士将移动 (a_1, b_1) 执行 $(Xs_1 + ka_2)$ 次，移动 (a_2, b_2) 执行 $(Xs_2 - ka_1)$ 次之后，将从原点到达点 (X, Y) 。至此，命题 1 得证。

问题二

回到猜想

- 要证明原猜想，只要把它转化为命题 **1** 即可。
- 命题 **1** 与原猜想的**本质区别**在于，它增加了一个条件： **a_1** 和 **a_2** 互质， **b_1** 和 **b_2** 互质。

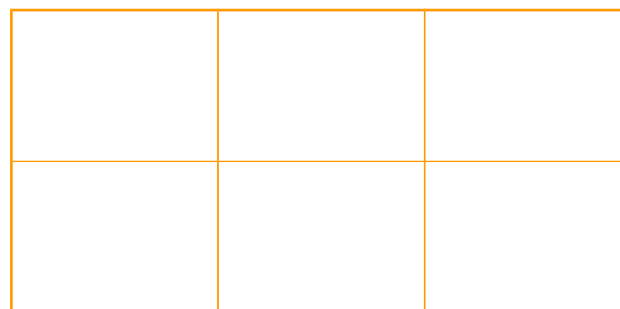
问题二 回到猜想

如果 \mathbf{a}_1 和 \mathbf{a}_2 不互质， \mathbf{b}_1 和 \mathbf{b}_2 不互质

令 $\mathbf{d}_1 = (\mathbf{a}_1, \mathbf{a}_2)$ ， $\mathbf{d}_2 = (\mathbf{b}_1, \mathbf{b}_2)$ 。

命题 1 中的一个格子等价于猜想中的 $\mathbf{d}_1 \times \mathbf{d}_2$ 个格子。

以 $\mathbf{d}_1 = 3$, $\mathbf{d}_2 = 2$ 为例



问题二

思路回顾

- 考虑一维情况，得到猜想。
- 对猜想的一个特例（命题 **1**），给出构造性证明。
- 将特例推广到一般情况，证明猜想。

问题二 小结

- 有些题目条件与结果之间的联系不很明显，难以找到突破口，从比较原始的地方（比如本题中的一维情况），能够看清楚问题；或者将简单情形（本题中的命题 **1**）作为一面镜子来为一般情形（本题中的猜想）造成某种对比，从对比中发现两种情形最本质的不同之处，再对症下药，求得问题的彻底解决。

问题二

小结(续)

- 特别地，在某些构造类的问题中（本题中的命题 **1**），可行的情况很多，但我们只需要一个。此时只需要考虑某种较为简单的情况，就能迅速有效地解决问题。这样的例子有很多，比如论文中的例 **3**。（**Baltic OI 2005 Polygon**）

总结

- “万物皆数” ----- 毕达哥拉斯
- 从特殊情况考虑，事实上是一种数学思想，它能够帮助我们在各个方面，包括信息学竞赛中，更好地认识事物的本质。
- 认识了事物的本质，我们就能够比较轻松地解决问题。而且，还能够在解决特殊情况的过程中，得到有益的启示，为处理一般情况提供对策，这一点比解决一个问题本身更为重要。