

# 知识精炼（一）



主讲人：邓哲也



# P0J3660 Cow Contest

- 有  $n$  头奶牛参加编程比赛，每一头奶牛的编程水平都各不相同。
- 如果奶牛  $A$  的水平比奶牛  $B$  的水平高，那么  $A$  永远会战胜  $B$ 。
- 现在比赛已经进行了  $m$  轮，你已经知道了在第  $i$  轮中，奶牛  $a[i]$  战胜了奶牛  $b[i]$ ，现在你想知道有几头奶牛的名次已经确定了。
- $n \leq 100$ ,  $m \leq 4500$ .
- 样例（输出 2）

```
5 5
4 3
4 2
3 2
1 2
2 5
```

# P0J3660 Cow Contest

- 很显然，如果 A 能打败 B， B 能打败 C，那么 A 就能打败 C。
- 比如如果能确定一头牛是第一名，那它一定可以打败所有的牛。
- 如果 A 能打败 B，就从 A 往 B 连一条边。
- 那么从第一名的牛出发，一定能遍历全部的牛。

# POJ3660 Cow Contest

- 如果能确定第二名的牛，那一定有一头牛能走到他，其他的牛都能从这头牛出发走到。
- 同理，如果能确定第  $k$  名的牛，那一定有  $k - 1$  头牛能走到它，它出发能走到剩下的  $n - k$  头牛。
- 因此我们只需要处理出每两头牛之间是否可达的信息即可。
- 这也就是传递闭包。

# P0J3660 Cow Contest

- $f[i][j]$  表示  $i$  是否能走到  $j$ 。
- 运用 floyd 算法,  $f[i][j] = f[i][j] \text{ or } (f[i][k] \text{ and } f[k][j])$
- 然后对于第  $i$  个点, 枚举有几个点可以走到  $i$ , 几个点可以从  $i$  出发走到。
- 两者加起来若等于  $n - 1$ , 就说明名次确定了。
- 时间复杂度  $O(n^3)$ 。

下节课再见