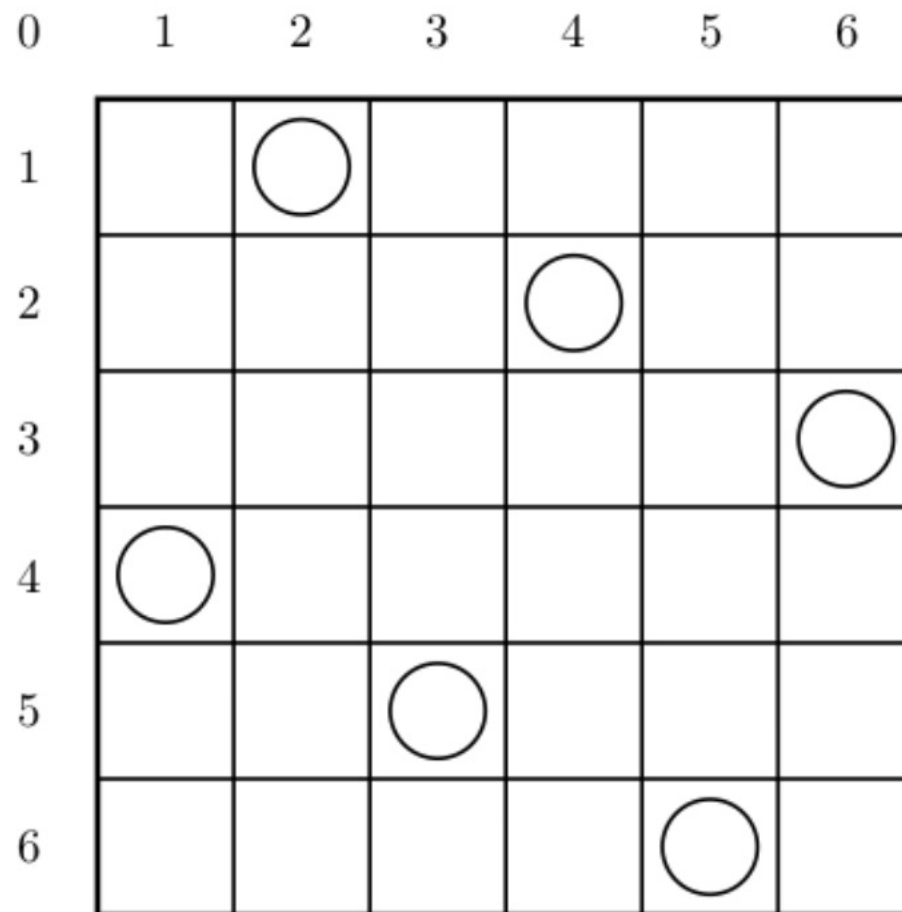


# 第一讲：八皇后问题 -P1219

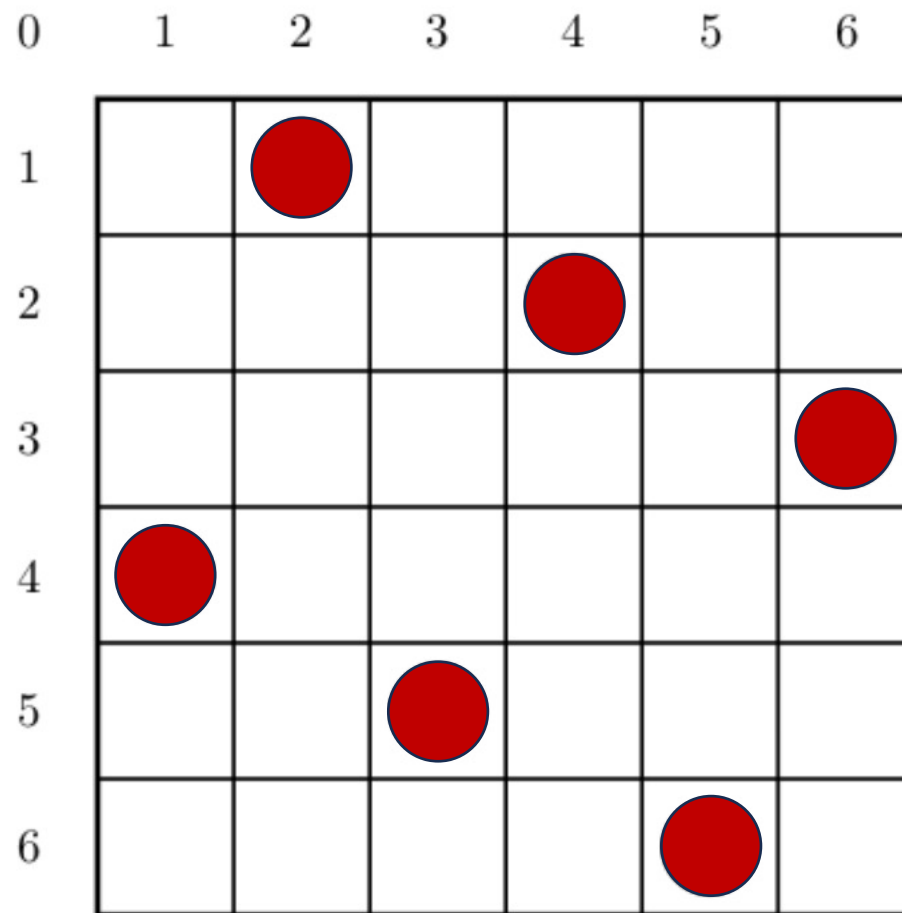
胡船长

初航我带你，远航靠自己

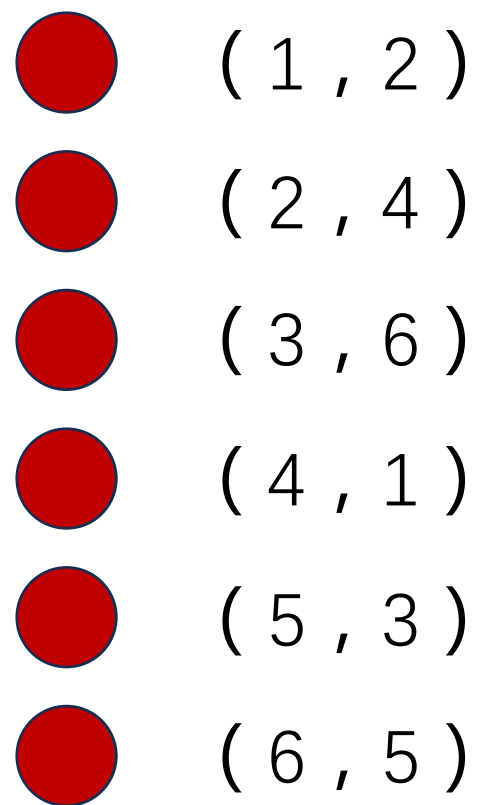
# 一、问题搜索树



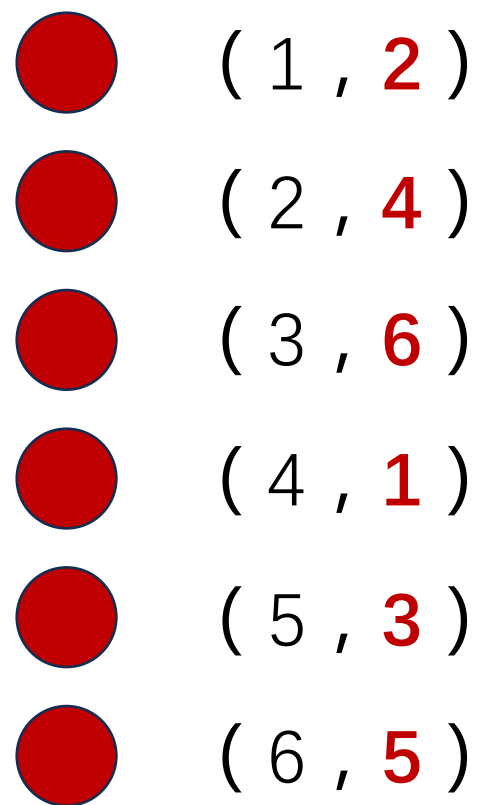
# 一、问题搜索树



# 一、问题搜索树



# 一、问题搜索树



# 一、问题搜索树

- ( 1 , 2 )
- ( 2 , 4 )
- ( 3 , 6 )
- ( 4 , 1 )
- ( 5 , 3 )
- ( 6 , 5 )

## 排列型枚举问题

## 二、优化1：状态表示

$$t = \begin{array}{ccccccc} & 6 & 5 & 4 & 3 & 2 & 1 & 0 \\ \hline 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{array}$$

# 二、优化1：状态表示

● ( 1 , 2 )

● ( 2 , 4 )

●

●

●

●

t =

6	5	4	3	2	1	0
1	1	0	1	0	1	0



### 三、优化2：快速枚举

-t & t

	6	5	4	3	2	1	0
t =	1	0	0	1	1	0	0

### 三、优化2：快速枚举

-t & t

	6	5	4	3	2	1	0
t =	1	0	0	1	1	0	0

	6	5	4	3	2	1	0
-t =	0	1	1	0	1	0	0

### 三、优化2：快速枚举

-t & t

t =

6	5	4	3	2	1	0
1	0	0	1	1	0	0

-t =

6	5	4	3	2	1	0
0	1	1	0	1	0	0

6	5	4	3	2	1	0
0	0	0	0	1	0	0

### 三、优化2：快速枚举

-t & t

t =

6	5	4	3	2	1	0
1	0	0	1	1	0	0

-t =

6	5	4	3	2	1	0
0	1	1	0	1	0	0

6	5	4	3	2	1	0
0	0	0	0	1	0	0

### 三、优化2：快速枚举

$$\underline{t \ -= \ (-t \ \& \ t)}$$

	6	5	4	3	2	1	0
$t =$	1	0	0	1	1	0	0

	6	5	4	3	2	1	0
$-t =$	0	1	1	0	1	0	0

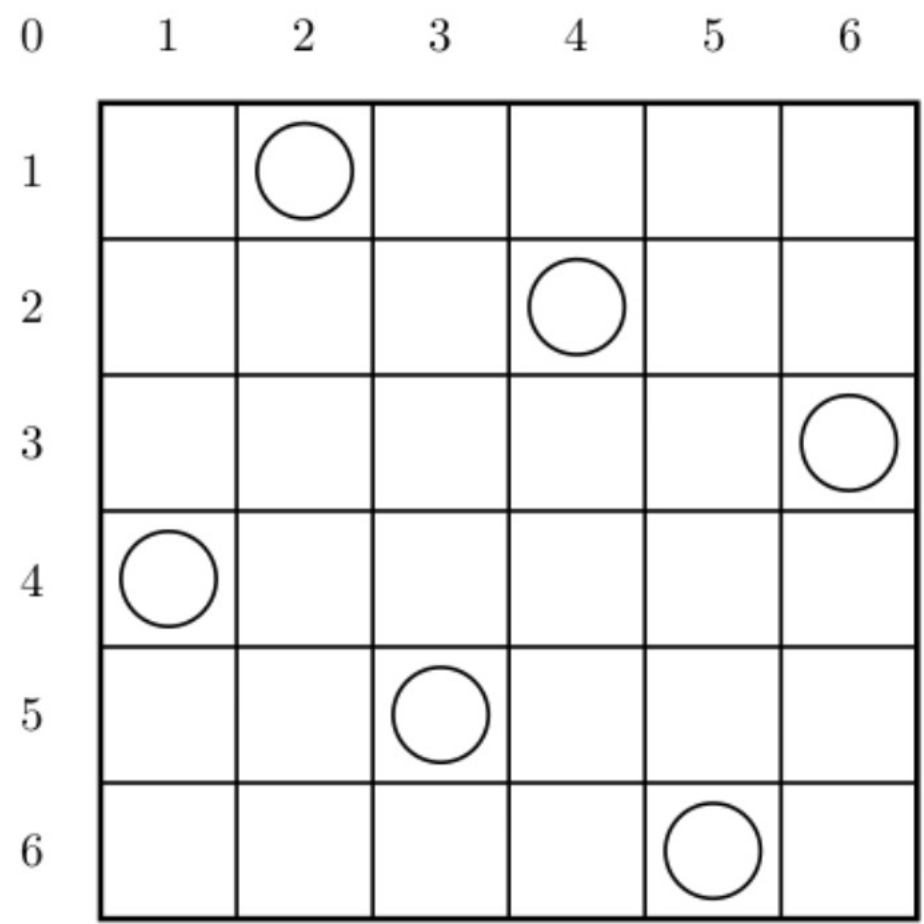
	6	5	4	3	2	1	0
	0	0	0	0	1	0	0

### 三、优化2：快速枚举

$$\underline{t \ -= \ (-t \ \& \ t)}$$

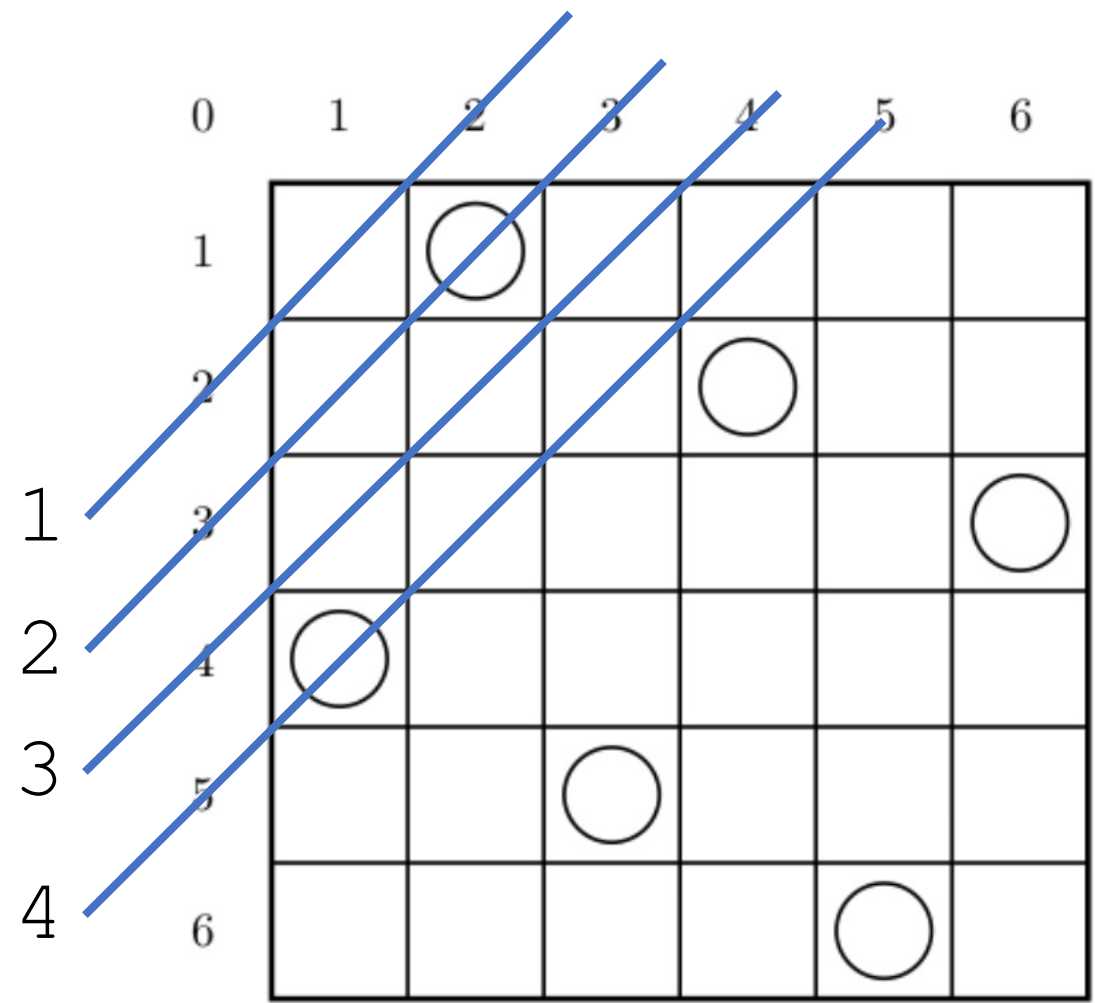
	6	5	4	3	2	1	0
t =	1	0	0	1	0	0	0

# 四、优化3：斜边表示



# 四、优化3：斜边表示

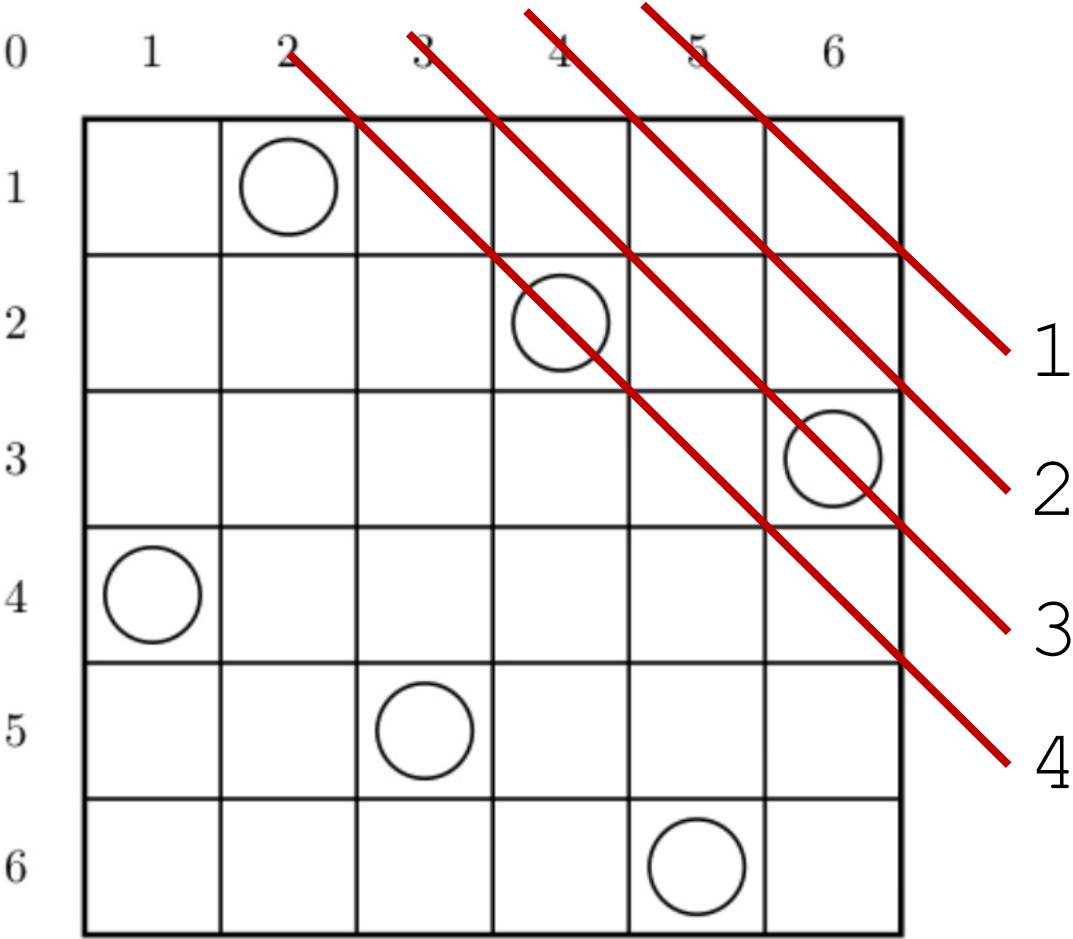
$i+j-1$



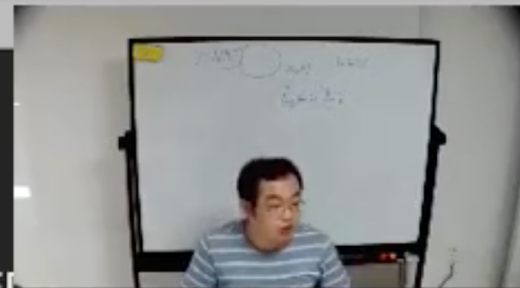


# 四、优化3：斜边表示

$i - j + n$

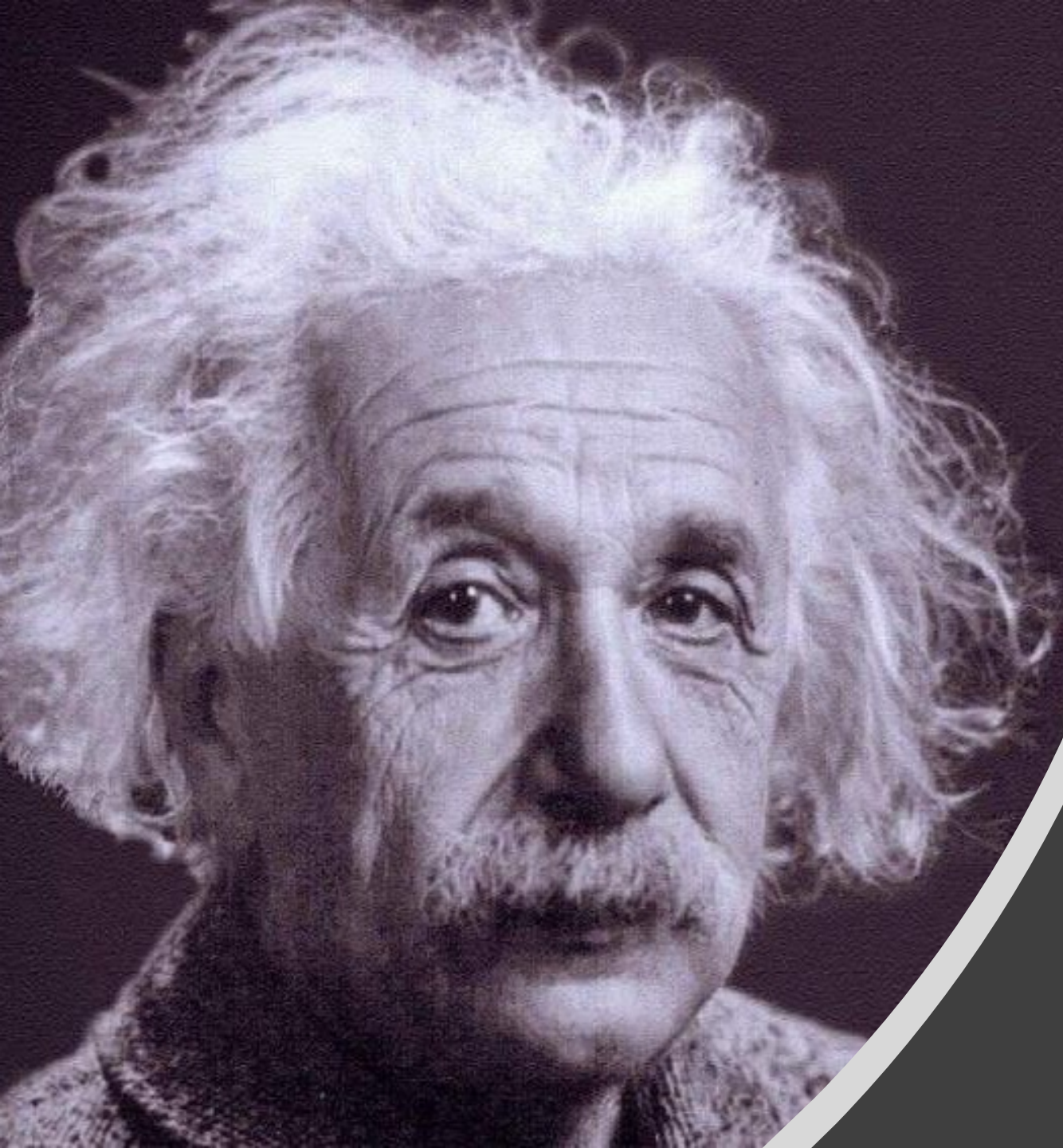


```
vim %1 bash %2 bash %3
39 }
40
41 Node *insert_maintain(Node *root) {
42     if (!hasRedChild(root)) return root;
43     if (root->lchild->color == RED && root->rchild->color == RED, {
44         if (!hasRedChild(root->lchild) && !hasRedChild(root->rchild)) return root;
45         root->color = RED;
46         root->lchild->color = root->rchild->color = BLACK;
47         return root;
48     }
49     if (root->lchild->color == RED) {
50         if (!hasRedChild(root->lchild)) return root;
51     }
52
53     } else {
54         if (!hasRedChild(root->rchild)) return root;
55     }
56 }
57
58
```



## 八皇后问题-P1219：代码演示

```
61 Node *__insert(Node *root, int key) {
62     if (root == NIL) return getNewNode(key);
```



为什么  
会出一样的题目？