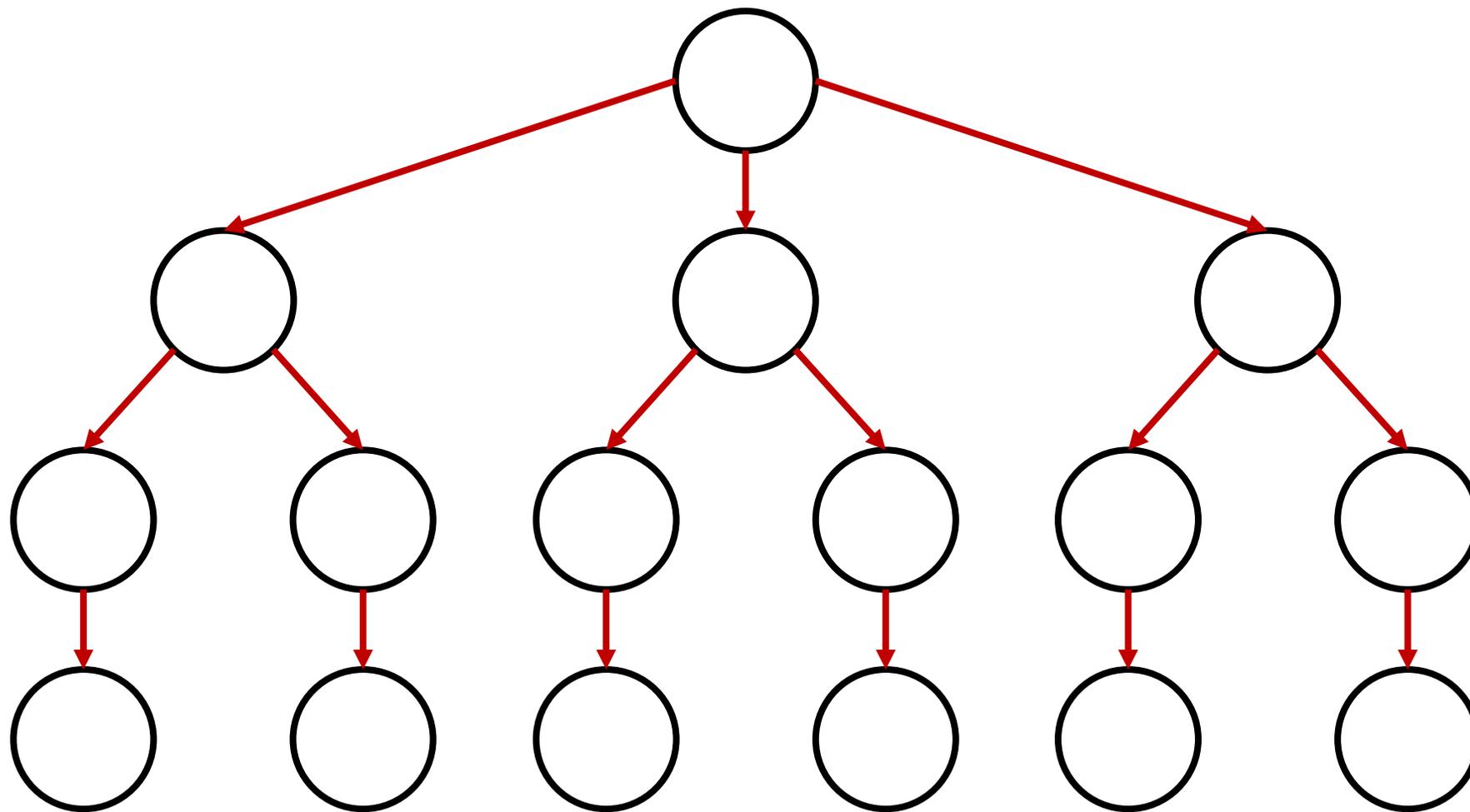


第四讲：马的遍历-P1443

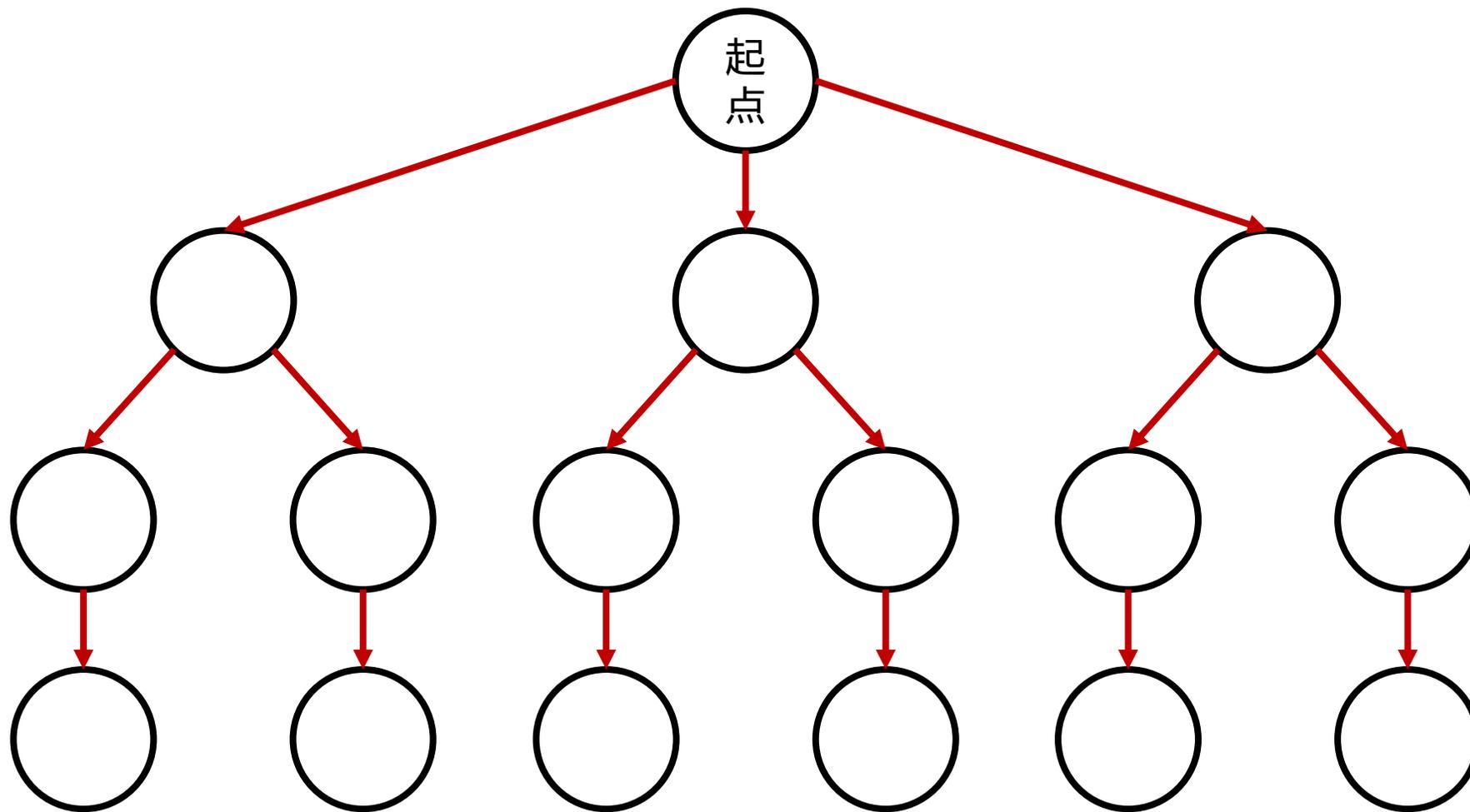
胡船长

初航我带你，远航靠自己

一、问题搜索树



一、问题搜索树



三、编程技巧：方向数组

| | | |
|------------|-----------|-----------|
| $(-1, -1)$ | $(-1, 0)$ | $(-1, 1)$ |
| $(0, -1)$ | (x, y) | $(0, 1)$ |
| $(1, -1)$ | $(1, 0)$ | $(1, 1)$ |

三、编程技巧：方向数组

| | dx | dy |
|--|--------|------|
| | (-1 , | -1) |
| | (0 , | -1) |
| | (1 , | -1) |
| | (-1 , | 0) |
| | (1 , | 0) |
| | (-1 , | 1) |
| | (0 , | 1) |
| | (1 , | 1) |

三、编程技巧：方向数组

| | | | | |
|--|--|--------|--|--|
| | | | | |
| | | | | |
| | | (x, y) | | |
| | | | | |
| | | | | |

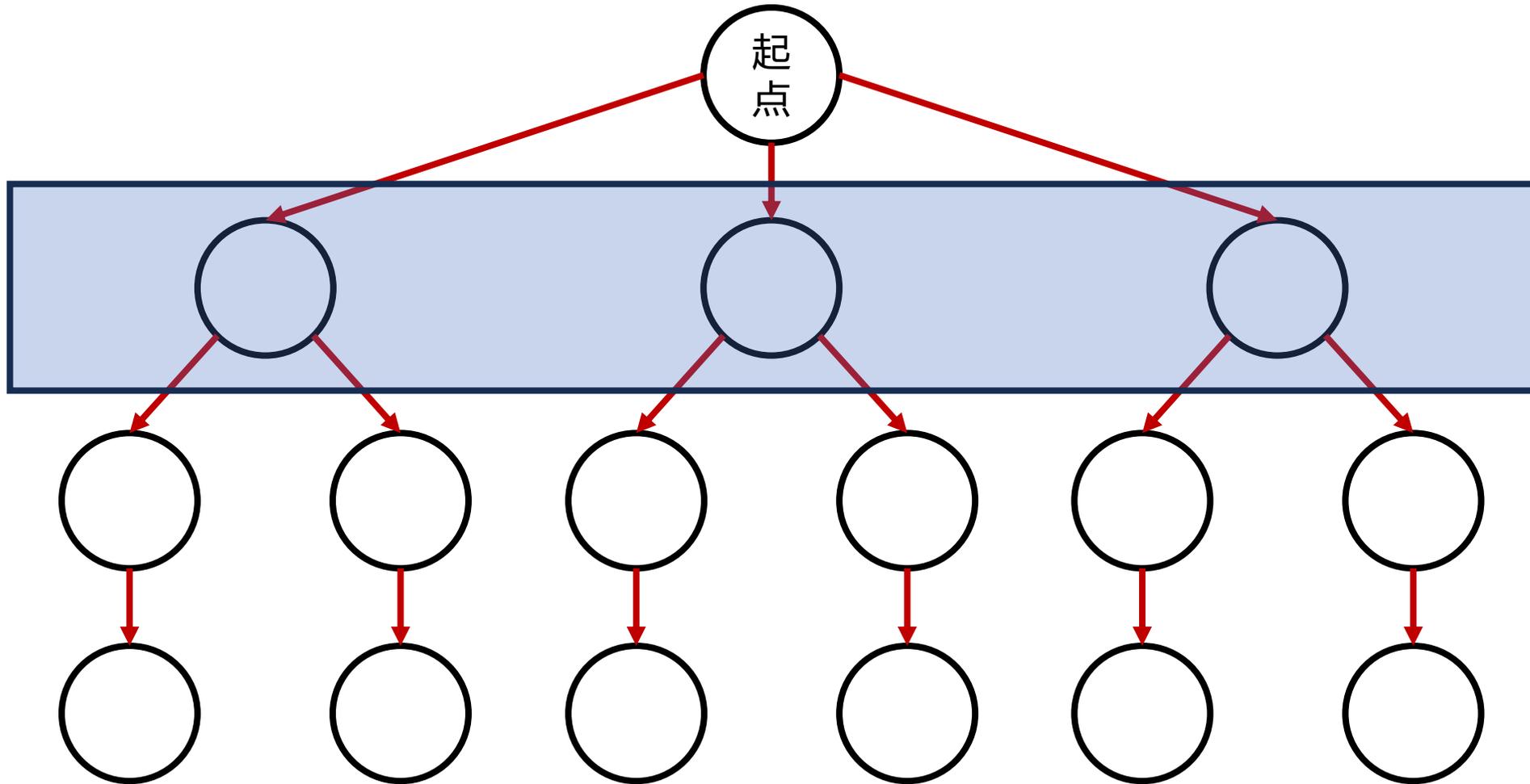
三、编程技巧：方向数组

| | | | | |
|---|---|--------|---|---|
| | ■ | | ■ | |
| ■ | | | | ■ |
| | | (x, y) | | |
| ■ | | | | ■ |
| | ■ | | ■ | |

三、编程技巧：方向数组

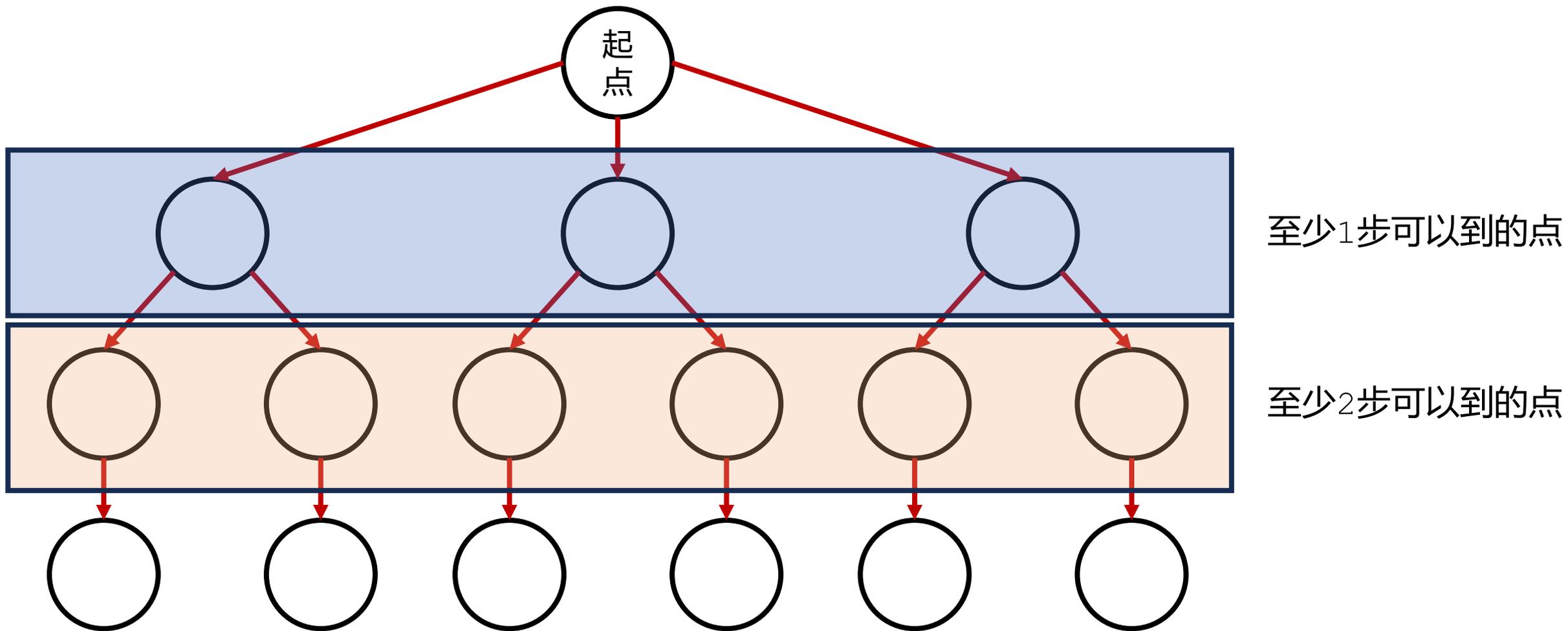
| | | | | |
|------------|------------|----------|-----------|-----------|
| | $(-2, -1)$ | | $(-2, 1)$ | |
| $(-1, -2)$ | | | | $(-1, 2)$ |
| | | (x, y) | | |
| $(1, -2)$ | | | | $(1, 2)$ |
| | $(2, -1)$ | | $(2, 1)$ | |

四、问题搜索树-BFS

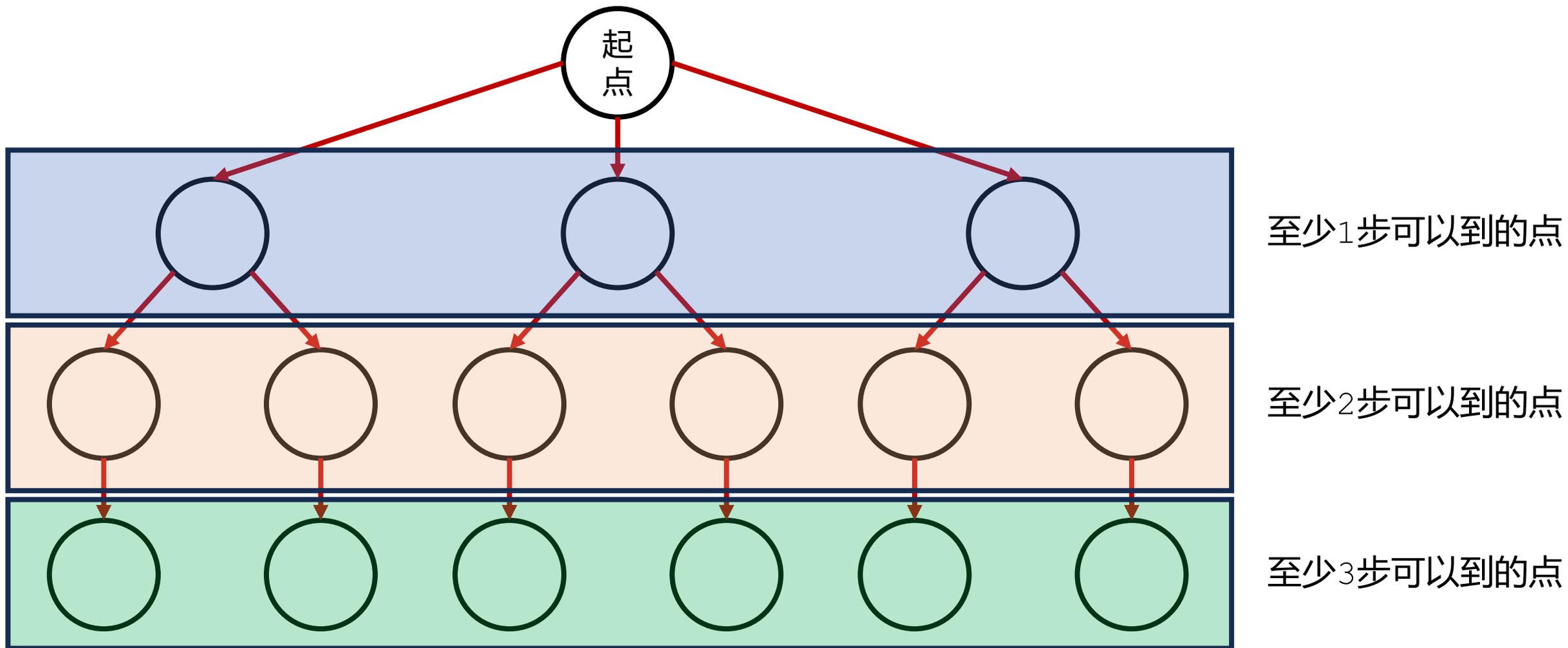


至少1步可以到的点

四、问题搜索树-BFS



四、问题搜索树-BFS



四、问题搜索树-BFS



四、问题搜索树-BFS

出现过的状态，不重复添加

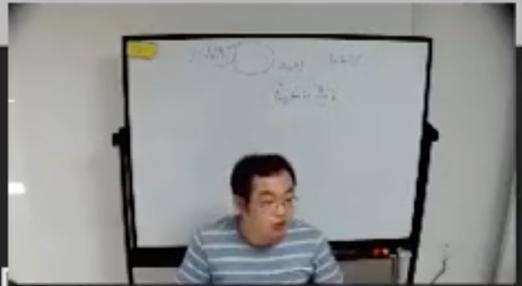


四、问题搜索树-BFS

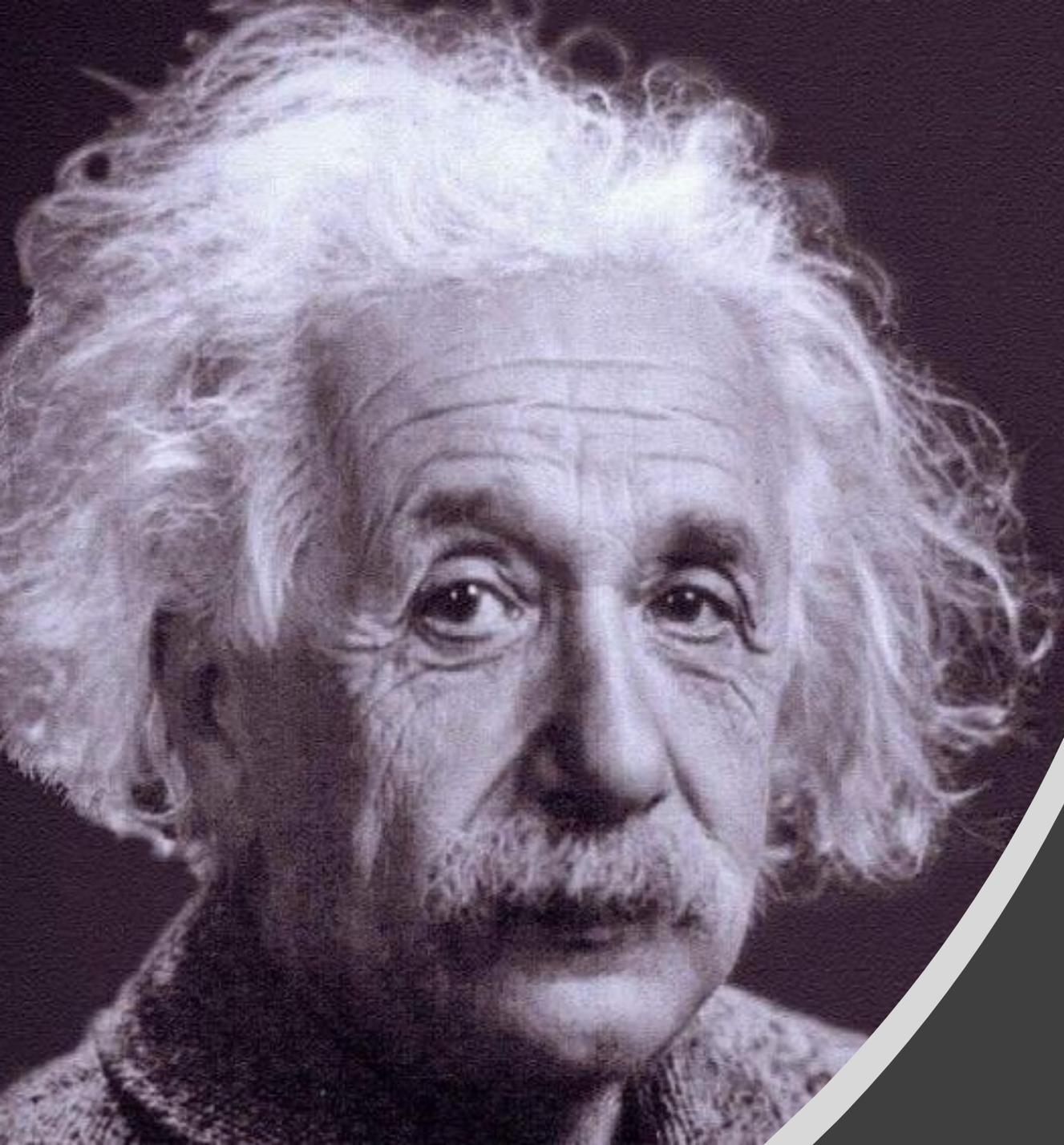
出现过的状态，不重复添加



```
1. vim
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     } else {
53         if (!hasRedChild(root->rchild)) return root;
54
55     }
56 }
57
58
59
60
61 Node *__insert(Node *root, int key) {
62     if (root == NIL) return getNewNode(key);
```



马的遍历-P1443：代码演示



为什么
会出一样的题目？