

错综复杂的大问题，我不断地去划分再划分

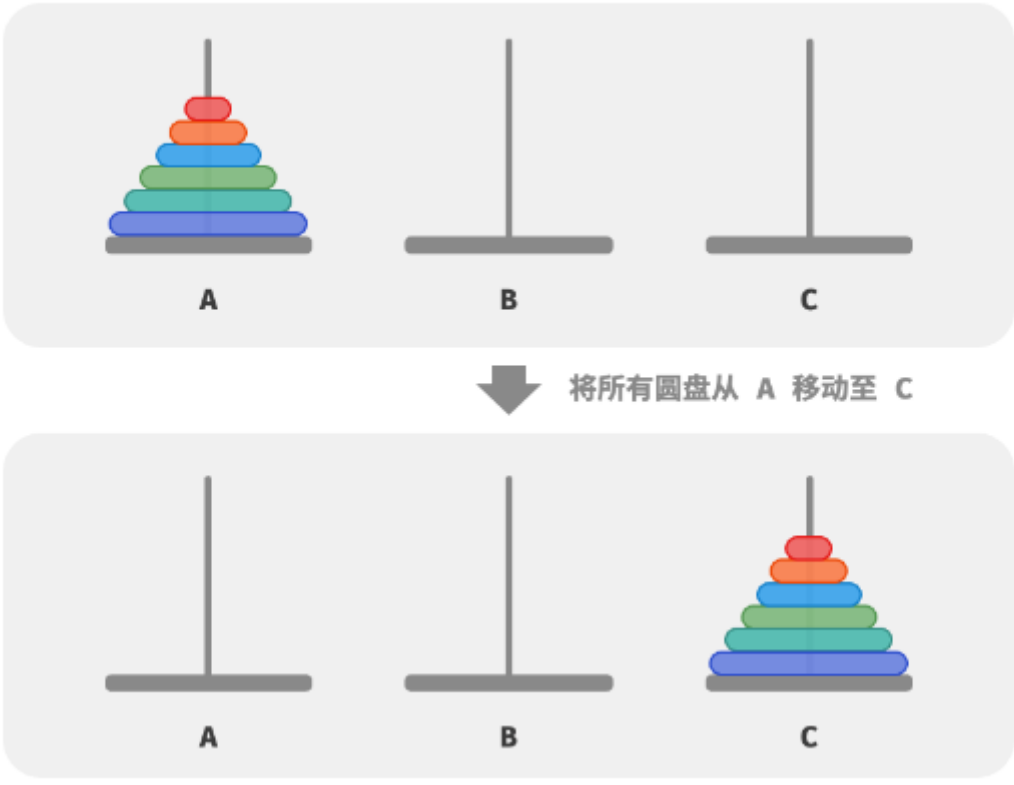
汉诺塔：n个圆盘移动只分为 n-1 和 1个圆盘的移动 但是对于 n-1 和 1 来说，src buf 不同 虽然 tar相同 src：初始柱子 buf：缓冲柱子 tar：目标柱子

```
void move(stack<int>& src, stack<int>& tar) {
    int temp = src.top();
    src.pop();
    tar.push(temp);
}

void dfs(int i, stack<int>& src, stack<int>& buf, stack<int>& tar) {
    if (i == 1) {
        move(src, tar);
        return;
    }

    dfs(i - 1, src, tar, buf);
    move(src, tar);
    dfs(i - 1, buf, src, tar);

    return;
}
```



至此，我们可总结出图 12-14 所示的解决汉诺塔问题的分治策略：将原问题 $f(n)$ 划分为两个子问题 $f(n - 1)$ 和一个子问题 $f(1)$ ，并按照以下顺序解决这三个子问题。

- 1. 将 $n - 1$ 个圆盘借助 C 从 A 移至 B。
- 2. 将剩余 1 个圆盘从 A 直接移至 C。
- 3. 将 $n - 1$ 个圆盘借助 A 从 B 移至 C。

对于这两个子问题 $f(n - 1)$ ，可以通过相同的方式进行递归划分，直至达到最小子问题 $f(1)$ 。而 $f(1)$ 的解是已知的，只需一次移动操作即可。

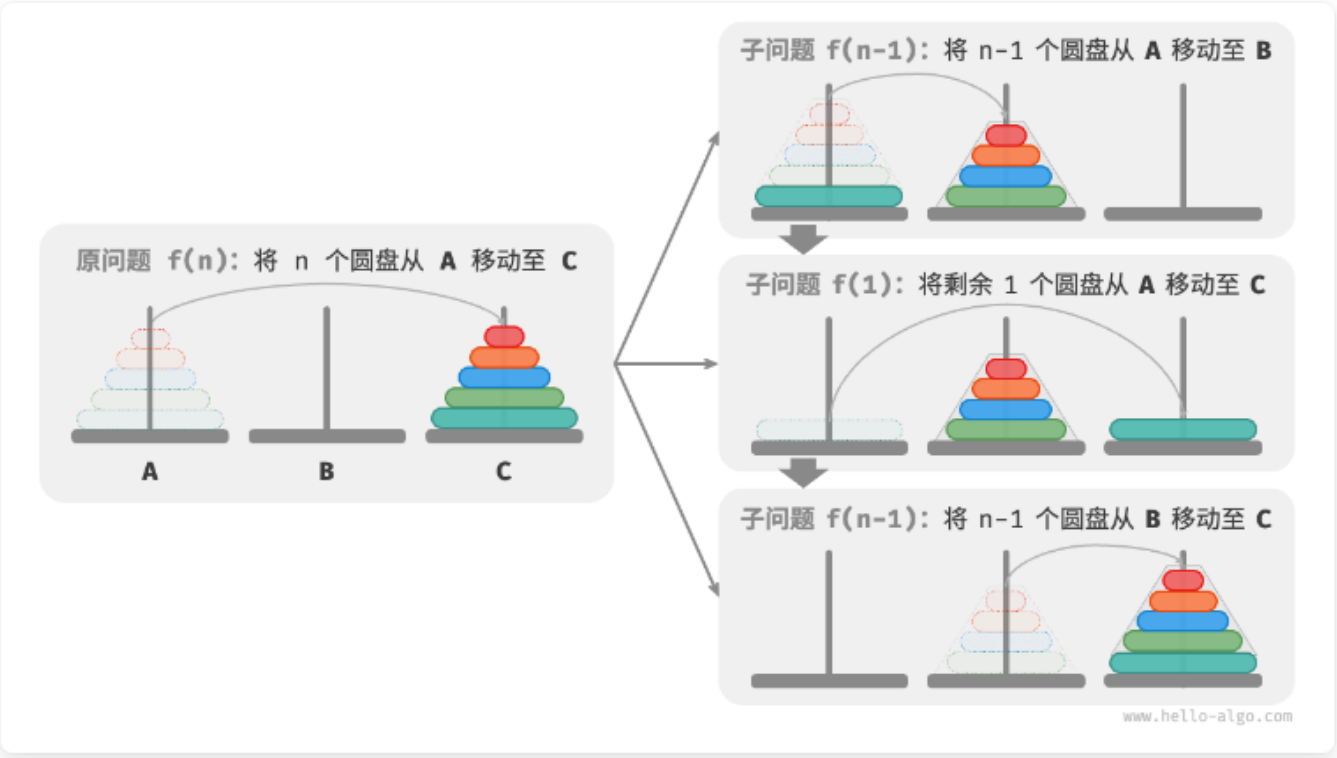


图 12-14 解决汉诺塔问题的分治策略