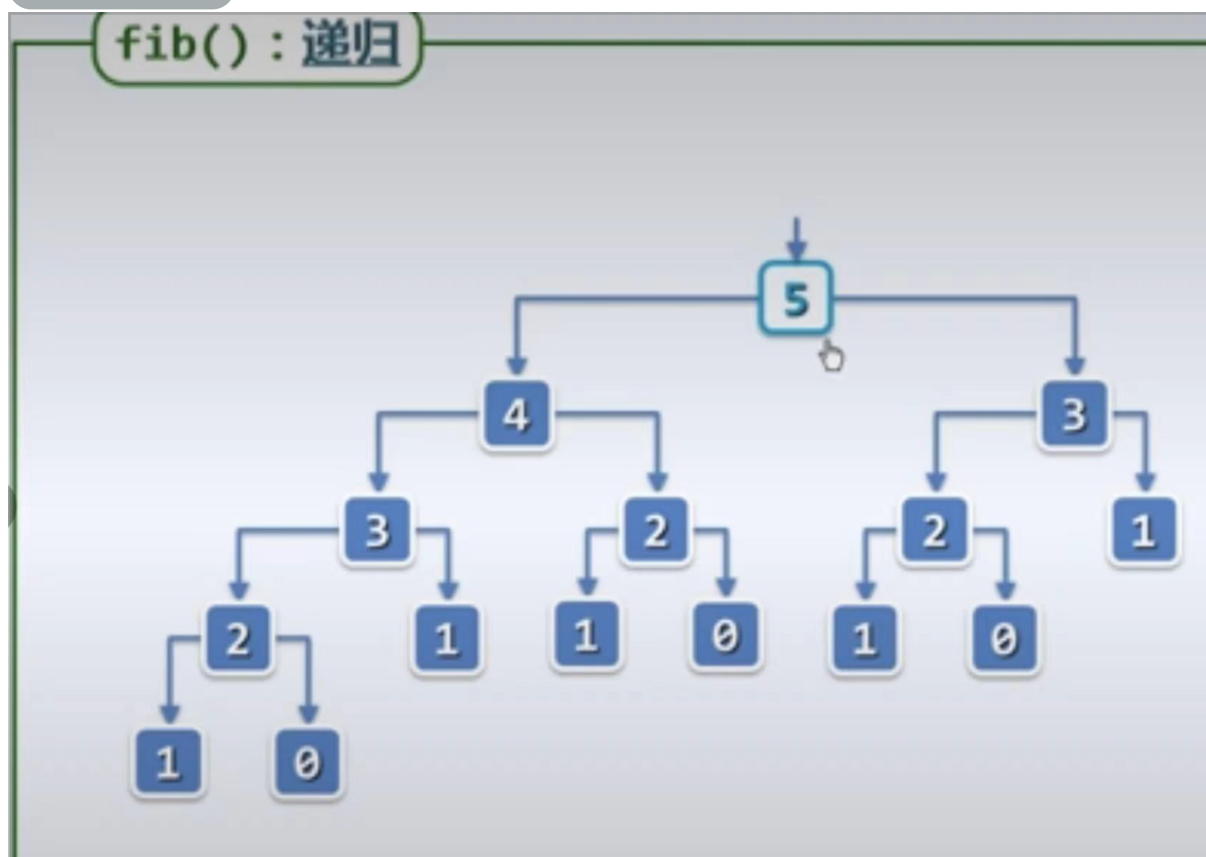


01-F-4 递归跟踪 & 01-F-5 回归迭代

#数据结构邓神



我们会发现计算非常多的重复!!!

导致计算效率奇低，大部分算力都被浪费了！

低效的原因在于：各递归实例均被大量重复的调用。

如何变成高效： 让每个实例计算一次

解决方法 A： (记忆：memoization)

将以计算过的实例的结果制表备查

解决方法 B： (动态规划：dynamic programing)

颠倒计算方向，由自顶向下，改为从底部向上迭代

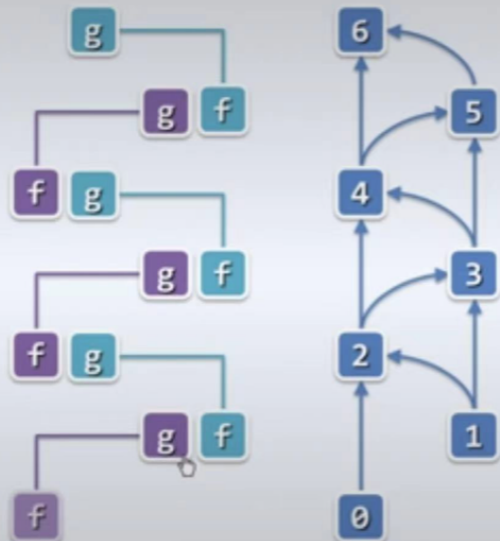
fib() : 迭代

❖ 解决方法A (记忆 : memoization)

将已计算过实例的结果制表备查

❖ 解决方法B (动态规划 : dynamic programming)

颠倒计算方向 : 由自顶而下递归, 为自底而上迭代



解决方法B 要比 A 更加节省空间和时间复杂度 (不用查表)

B

```
// 给定整数 n
int f = 0;
int g = 1;
while (0 < n--){
    g = g+f;
    f = g-f;
}
return g;
```