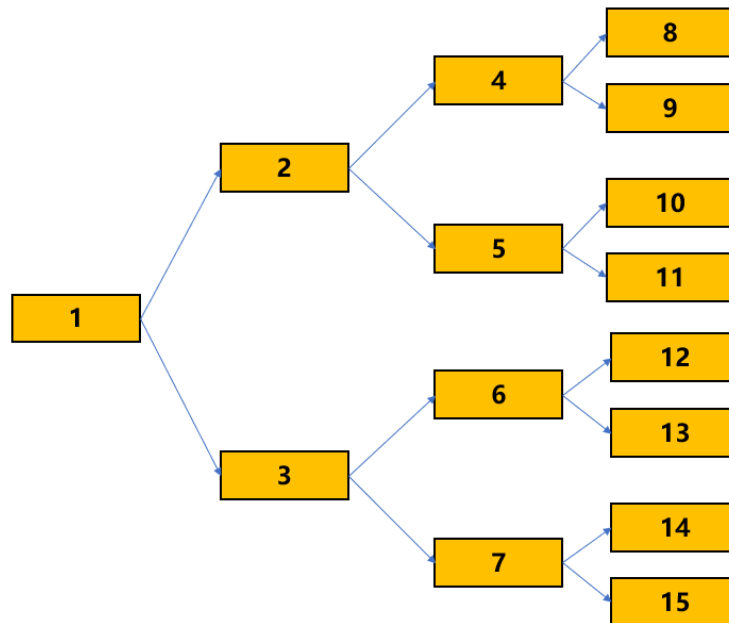


# 人工智能基础作业 2

张磊 2017K8009922027

3.15 考虑起始状态为 1、每个状态  $k$  都有两个后继  $2k$  和  $2k+1$  的状态空间。

a. 画出 1-15 的状态空间；



b. 假设目标状态为 11。请列出访问节点的顺序：宽度有限搜索、深度界限为 3 的深度受限搜索、迭代加深搜索。

1. 宽度优先：1,2,3,4,5,6,7,8,9,10,11,12,13,14,15;

2. 深度界限为 3 的深度受限搜索：1,2,4,5,3,6,7;

3. 迭代加深搜索（初始深度  $d == 1$ ）： 1;

1,2,3;

1,2,4,5,3,6,7;

1,2,4,8,9,5,10,11,3,6,12,13,7,14,15;

c. 双向搜索求解此问题有优势吗？两种方向的分支因子是多少？

双向搜索求解此问题没有优势，原因如下：

通常情况下，假设搜索一棵分支因子  $b$  的树，初始节点到目标节点的距离为  $d$ ，该算法的正向和反向搜索复杂度都是  $O(b^{\frac{d}{2}})$  (大  $O$  符号)，两者相加后远远小于普通的单项搜索算法（复杂度为  $O(b^d)$ ）。

但不幸的是，在这个问题中，树节点的分支因子为  $b == 2$ ，当  $d == 2$  时  $O(b^{\frac{d}{2}})$  与  $O(b^d)$  相等，因此，双向搜索对求解此类问题没有优势：

**d. 求解问题 c 是否需要对问题重新进行形式化成允许求解从状态 1 到一给定目标状态的问题时几乎没有搜索？**

因未读懂题意，所以暂时没有作答；

**e. 调用行动从  $k$  到左边的  $2k$  和右边的  $2k+1$ 。你能找到完全不用搜索就能求得解的算法吗？**

将所需进行搜索的数据按照从 1 到  $n$  的顺序，顺序存放到数组中，即可根据二叉树的性质直接计算得出目标结点；

**选做：八码问题**