二叉树的深搜和广搜

为什么有广搜和深搜，问你个问题给你一个二叉树，给你一坨数据问这些数据中的比如A在不在二叉树中，B在不在？你得告诉我它是在里面还是不在里面，这时候就需要一种逻辑。

搜索：

1. 深度优先搜索（往外走）和前序遍历的顺序是一样的
   1. 能往下找就往下找：先找A没有再找C再找F再找F之后的如果没有那么回到C再找G，逐级往下找，更适合去探索未知，同样的CSDN找一篇文章那么会跳出CSDN。
2. 广度优先搜索（局域搜索）
   1. 按层找，先找A没有找C还没有然后横着找，找到B没有找F没有找G，一层一层的找，从左往右，更适合探索局域，比如我找CSDN里面的一篇文章，广度寻找那就在CSDN里面找。

G在不在二叉树中？

A

B

C

E

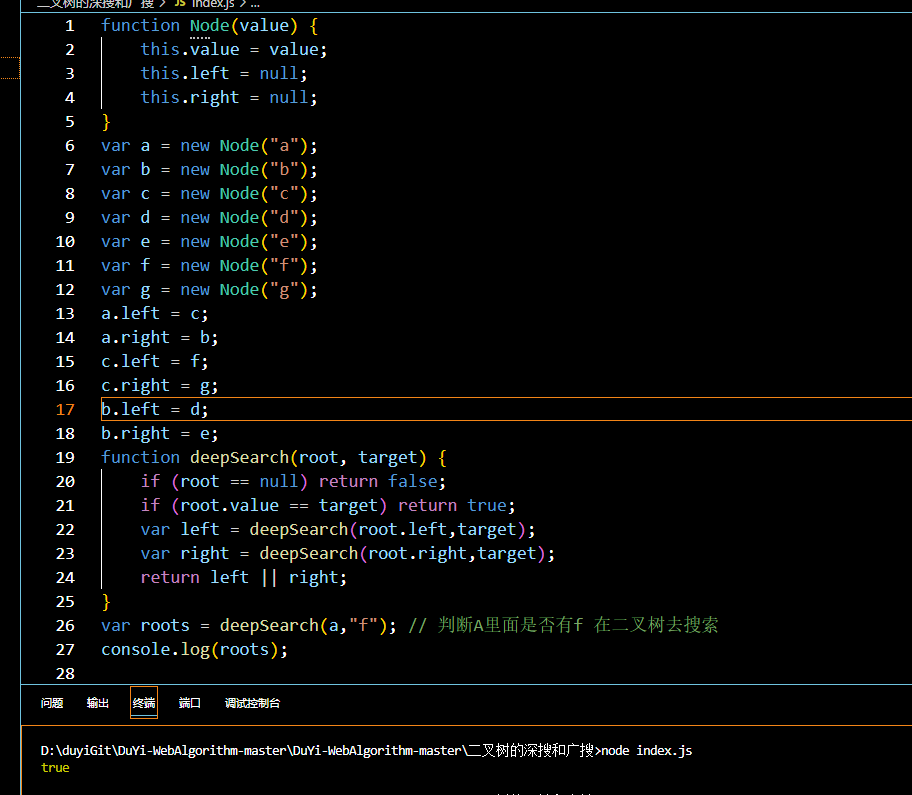
D

G

F

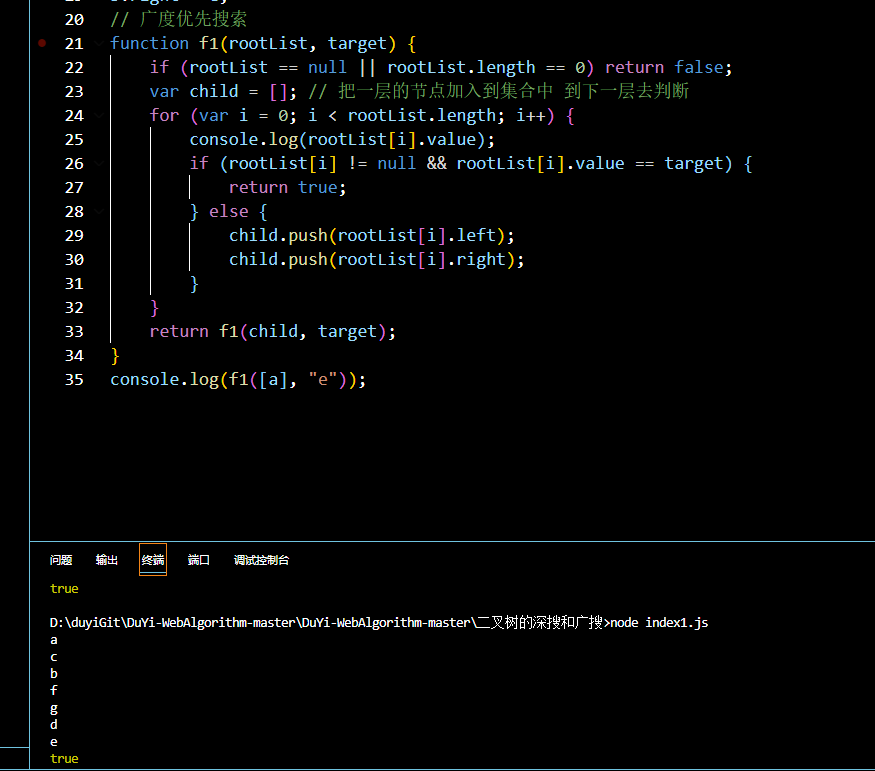
深搜广搜的代码实现：

深度优先的搜索的实现：在二叉树中找某一个数据有就给我返回true否则就false。



二叉树的广度优先搜索:

算法：把当前的这一层进行一个判断，如果不行就把当前层的子节点都打包到集合中再去下一层进行判断直到找到正确的值为止。



二叉树的搜索分支:

1. 树的搜索
2. 图的搜索
3. 爬虫的逻辑
4. 搜索引擎的爬虫算法