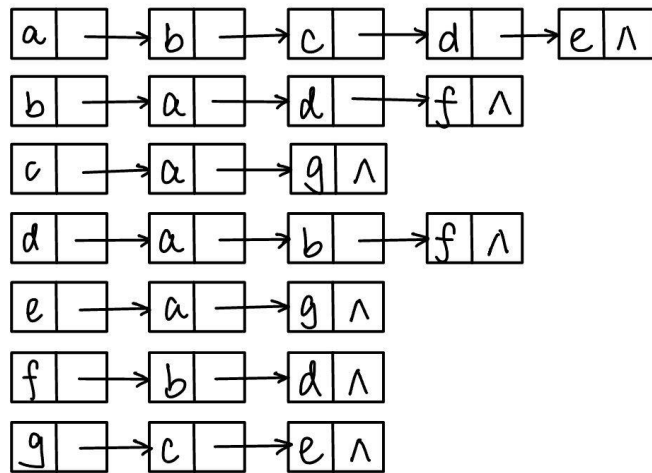


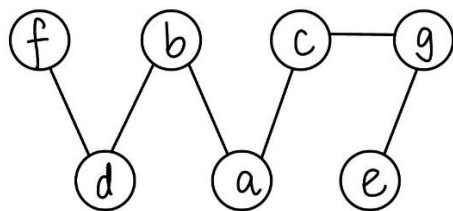
1. 平均情况下，顺序查找的时间复杂度为 $O(n/2)$ ；折半查找的时间复杂度为 $O(\log n)$ 。
 $(10000000/2)/\log_2(10000000) \approx 215021$
 因此，折半查找比顺序查找快大约 215021 倍。
2. 每一个探险家若想到达对岸，都需要船的 4 次横渡：2 名水手从起点到终点；1 名水手从终点到起点；1 名探险家从起点到终点；1 名水手从终点到起点。
 因此，这条船需要在岸与岸之间横渡 $4N$ 次。
3. 前序：abdcef
 中序：dbaecf
 后序：dbefca
4. ALGORITHM exhaustiveSearch(G,k)
 {
 //查找 G 是否包含大小为 k 的完备子图
 //穷举所有 k 个节点的组合，并判断每个组合是否为完备图
 //Input: 图 G 的信息, k
 //Output: true/false
 n=G 中的节点个数
 //该函数判断指定节点集合是否形成完全图
 Function isComplete(nodes)
 {
 For $i \leftarrow 1$ to $k-1$ do
 For $j \leftarrow i+1$ to k do
 If nodes[i]与 nodes[j]无边
 Return false
 Return true
 }
 //该函数生成所有可能的 k 个节点组合，并逐个检查
 Function generateCombinationsAndCheck(currentNodes, start)
 {
 If length(currentNodes)=k
 If isComplete(currentNodes)
 Return true
 Else
 Return false
 For $i \leftarrow start$ to n do
 把 node i 加入到 currentNodes 中
 If generateCombinationsAndCheck(currentNodes, i+1)
 Return true
 把 node i 从 currentNodes 中移除
 Return false
 }
 //开始生成所有组合并检查
 Return generateCombinationsAndCheck(currentNodes, 1)
 }
 }
5. a. 邻接矩阵:

0111100
 1001010
 1000001
 1100010
 1000001
 0101000
 0010100

邻接链表:



b. 深度优先查找树:



入栈顺序: abdfcge

出栈顺序: fdbegca