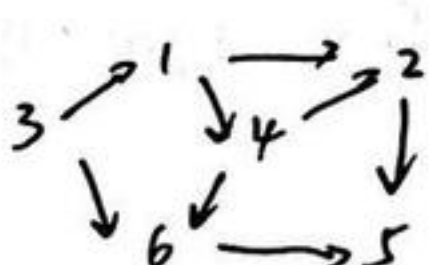
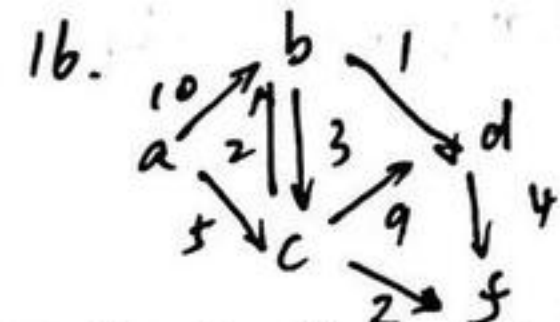


1. 长度为7的散列表,  $H(k) = k \bmod 7$ , 线性探测重哈希冲突, 22, 43, 15 插入后散列表元素查找3个关键字, 平均比较次数 ( ) 3 / 1.6 / 2 / 1.5
2. 3个结点的二叉树有 ( ) 种形态 7 / 3 / 5 / 2
3. 求前缀和问题: 最优子结构且子问题是重复求解, 采用 ( ) 策略? 贪心/分治/回溯/动态规划
4. 以下关于二叉树的叙述:
- A. 从左到右同一层的结点, 关键字有序排列
  - B. 对树前、中、后序遍历都得到关键字有序排列
  - C. 含n个结点的二叉树总层数为  $\lfloor \log_2 n \rfloor + 1$
  - D. 根到任意叶结点的路径上关键字有序排列
5. 元素移动次数与关键字初始排列次序无关的是 ( ) 快速/插入/计数/冒泡
6. for  $k=1; k \leq n; k++$  求时间复杂度 ( )  $\theta(n \log_2 n) / \theta(\log_2 n) / \theta(n^2) / \theta(n)$   
 for  $k=1; i \leq n; i++$   
 count++
7. 快排最适合的序列存储方式是 ( ) 顺序/链式/散列/索引
8.  可能的拓扑序列是 ( ) 314265 / 314256 / 312465 / 312456
9. Horspool 算法, 查找 "10000" (在10000个构成的文本中), 需进行 ( ) 比较 1992 / 4920 / 996 / 498
10. 关于AVL树和2-3树 ( ) 不正确
- A. 2-3树  $\leq$  AVL树 (高度)
  - B. 2-3树叶节点处于同一层
  - C. 查找相同关键字, 2-3树的平均比较次数  $\leq$  AVL树
  - D. AVL节点数  $\geq$  2-3树
11. 不可能是二分查找形成的 ( ) 18 20 50 45 / 50 20 45 18 / 50 45 20 18 / 18 50 20 45
12. 不是分支界限法常用搜索方式 ( ) 深度优先/最小耗费优先/广度优先/最大效益优先
13. 关于稳定婚姻问题
- A. 平凡下界即稳定下界
  - B. 结果与开始时的女/男选择无关
  - C. 唯一存在
  - D. 是女士最差的匹配
14. 前序: 123456 中序: 324165 求后序 ( ) 326154 / 342651 / 324651 / 364251
15. 25, 13, 10, 12, 9 构成最大堆, 插入18后重新调整为最大堆, 调整中元素的最少比较次数? 4 / 2 / 5 / 1



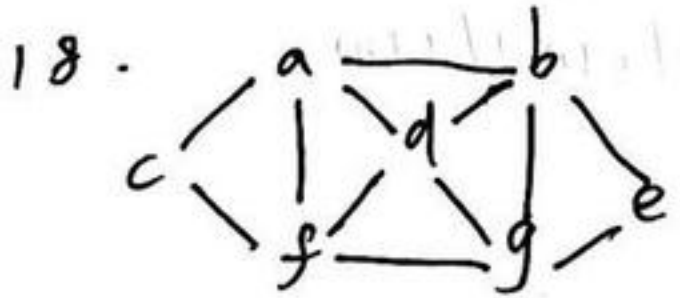
求  $a \rightarrow e$  最短权重路径

- 1. 算法名称
- 2. 主要思想
- 3. 应用于本例的执行细节并给出结果

17.  $N$  个人, 每人买一张票, 可单独买耗时  $T[i]$ , 也可以给后面的人一起买耗时  $R[i]$ , 有:

$$T[i] < R[i] < T[i] + T[i+1]$$

- (1) 给出  $T[i]$  ( $i=1, 2, \dots, N$ ),  $R[i]$  ( $i=1, 2, \dots, N-1$ ), 求最佳买票方式使时间最短, 用代码实现
- (2) 分析时间效率与空间效率



用回溯法求一条哈密顿回路, 标出搜索过程, 以  $a$  为起, 以字母顺序确定访问先后

19. 阿里  $\xrightarrow{\text{offer}}$  ABC 腾讯  $\xrightarrow{\text{offer}}$  ABDE 百度  $\xrightarrow{\text{offer}}$  C 网易  $\xrightarrow{\text{offer}}$  AC 新浪  $\xrightarrow{\text{offer}}$  ...  
每家公司最多招 1 人

- (1) 一般化上述问题为  $N$  人  $M$  公司, 设计分配算法并描述
- (2) 针对上例给出结果

20. 小明提出用新的分治算法求最小生成树, 给定无权重无向图  $G=(V, E)$ , 首先将点集  $V$  划分为  $V_1$  和  $V_2$ ,  $|V_1| - |V_2| \leq 1$ ; 然后在  $G_1=(V_1, E_1)$  和  $G_2=(V_2, E_2)$  上分别递归求最小生成树问题, 最后从  $E$  中选择一条连接  $V_1, V_2$  的最小权边, 合并生成完整的树。

- (1) 判断是否总能正确地计算出最小生成树?
- (2) 论证你的判断。