

哈工大2013秋数据结构期末试题(含答案)

原创 dream or nightmare 于 2020-11-18 22:05:32 发布 阅读量2.2k 收藏 77 点赞数 18

分类专栏: 数据结构 文章标签: 哈工大数据结构期末试题 2013秋数据结构期末试题 哈工大数据结构 数据结构期末试题 数据结构

数据结构 专栏收录该内容

31 订阅 13 篇文章

- 哈工大2013秋数据结构期末试题
- 原题
- 答案
- 其他年份的题

原题

主管领导审核签字

哈尔滨工业大学 2013-2014 学年 秋 季学期

数据结构与算法 试 题

题号	一	二	三	四	五	六	七	八	九	十	总分
得分											

一、单项选择题：1~10 小题，每小题 2 分，共 20 分。在每小题给出的四个选项中，请选出最符合题目要求的一项。

1. 设 n 是自然数，程序段: "for(int x=1; x<=n; x*=5);"的时间复杂度为()。

A. $O(1)$ B. $O(\log_2 n)$ C. $O(n)$ D. $O(n^2)$

2. 对于 n 个元素组成的线性表，建立一个有序单链表的时间复杂度是 ()。

A. $O(\log_2 n)$ B. $O(n)$ C. $O(n\log_2 n)$ D. $O(n^2)$

3. 已知一棵度为 3 的树，有 2 个度为 1 的结点，3 个度为 2 的结点，4 个度为 3 的结点，则该树中共有 () 个结点。

A. 12 B. 14 C. 21 D. 23

4. 设森林中有 4 棵树，树中结点的个数依次为 n_1 、 n_2 、 n_3 、 n_4 ，则把森林转换成二叉树后，其根结点的左子树上有 () 个结点。

A. n_1-1 B. n_1 C. $n_1+n_2+n_3$ D. $n_2+n_3+n_4$

5. 有 5 个字符，根据其使用的频率设计对应的哈夫曼编码，() 不可能是哈夫曼编码。

A. 000,001,010,011,1 B. 0000,0001,001,01,1

C. 000,001,01,10,11 D. 00,100,101,110,111

6. 关键路径是 AOE 网中 ()。

A. 从源点到汇点的最短路径 B. 从源点到汇点的最长路径

C. 从源点到汇点的最短的回路 D. 从源点到汇点的最长的回路

第 1 页 (共 3 页)

https://blog.csdn.net/weixin_45406155

dream or nightmare

关注

7. 含 n 个顶点的连通图中的任意一条简单路径, 其长度不可能超过 ()。
- A. $n/2$ B. $n-1$ C. n D. $n+1$
8. 对数据序列 (15, 9, 7, 8, 20, 4, 1) 进行排序, 进行一趟 (遍) 后数据序列变为 (1, 9, 4, 8, 20, 7, 15), 则采用的是 () 排序。
- A. 气泡排序 B. 直接插入排序 C. 希尔排序 D. 堆排序
9. 下列各组排序方法中, 比较次数与待排序记录的初始状态无关的是 () 组。
- A. 插入排序和快速排序 B. 归并排序和快速排序
- C. 选择排序和归并排序 D. 插入排序和归并排序
10. 下列排序算法中, 其中 () 组排序算法都是稳定的。
- A. 堆排序和气泡排序 B. 快速排序和堆排序
- C. 直接选择排序和归并排序 D. 归并排序和气泡排序

二、填空题: 11~15 小题, 每小题 2 分, 共 10 分。把符合要求的答案填写在下划线处。

11. 算术表达式 $a+(b-c)*d$ 的前缀表达式为 _____, 后缀表达式为 _____。
12. 具有 n 个顶点的有向图若是强连通的, 最多有 _____ 边, 最少有 _____ 条边。
13. 高为 $h=6$ 的 AVL 树最多有 _____ 个结点, 最少有 _____ 个结点。
14. 高为 $h=5$ 的 3 阶 B-树, 最多有 _____ 个关键字, 最少有 _____ 个关键字。
15. 在具有 11 个元素的有序数组中查找一个指定元素, 若采用二分 (折半) 查找法, 最多需要进行 _____ 次比较, 最少需要进行 _____ 比较。

三、简答题: (16~17 小题, 每小题 8 分, 共 16 分)

16. Dijkstra 算法和 Floyd 算法都可以用于求解带权有向图的最短路径问题。请回答以下问题:

- (1) 为什么 Dijkstra 算法用于边带负权值的图时, 结果会不正确? 请举例说明。(3 分)
- (2) Floyd 算法允许图中的边带负权值, 但要求图中不能包含权值总和为负值的回路 (称为负权值回路)。请举例说明边带负权值的实际意义。(2 分)
- (3) 为什么 Floyd 算法要求图中不能包含负权值回路? 请举例说明。(3 分)

第 2 页 (共 3 页)

https://blog.csdn.net/weixin_45406155



dream or nightmare

关注

姓名: 学号: 院: 系: 封 线

17. 已知 12 个初始归并段的长度分别为 30, 44, 8, 6, 3, 20, 60, 18, 9, 62, 68, 85。现要做 4 路外部归并排序, 请回答以下问题:

(1) 试画出表示归并过程的最佳归并树; (6分)

(2) 计算该归并树的带权路径长度 WPL 和访问外存的次数。 (2分)

四、算法设计题: (18~19 小题, 每小题 12 分, 共 24 分。按以下要求设计算法: (1)给出算法的基本设计思想; (2)根据设计思想, 采用 C 或 C++或 JAVA 语言描述算法。 (3)说明你所设计算法的时间复杂度和空间复杂度)

18. 已知的字符串只包含数字 0 至 9。请设计一个算法 FirstNotRepeatingChar, 在长度为 n 字符串中找到第一个只出现一次的数字。如输入 01027243575, 则输出 1。要求时间复杂度为 $\Theta(n)$ 。

19. AVL 树的结点结构为: `lchild key size rchild`, 其中, key 为结点的关键字, lchild 和 rchild 分别为指向左、右子树的指针, size 为以该结点为根的 AVL 树的结点个数。请设计一个算法 FindKthKey 查找 AVL 树中第 k 小的关键字所在的结点。要求: 时间复杂度为 $\Theta(\log_2 n)$, 其中 n 为 AVL 树的结点个数。

第 3 页 (共 3 页)

https://blog.csdn.net/weixin_45406155

答案

参考答案

一、单项选择题：1~10 小题，每小题 2 分，共 20 分。

1. B 【详解】程序每执行一次，x 就乘以 5，但是 x 又是小于 n 的，所以 5 的 a 次方小于等于 n，所以时间复杂度为 $O(\log_5(n))$ ，满足条件是 B。
2. D 【详解】该算法需要将 n 个元素依次插入到有序单链表中，需要两重循环插入每个元素，所以选 D。
3. C 【详解】根据树的性质，设分支数为 B，结点数为 n，则 $B=n-1$ ，而 $B=2*1+3*2+4*3=20$ ， $n=21$ ，所以选 C。
4. A 【详解】根据森林与二叉树的转换规则，二叉树的根是第一株树的根，而第一株树的其他结点为左子树，所以左子树的结点数为 $n1-1$ ，选 A。
5. D 【详解】哈夫曼树的构造原理，所有结点的度要么为 0，要么为 2，而 D 选项的编码会出现度为 1 的结点，所以选 D。
6. B 【详解】根据关键路径定义，所以选 B。
7. C 【详解】简单环路也是简单路径，只是起点和终点重复，所以长度不会超过 n，选 C。
8. C 【详解】看序列数据分布，发现 1 与 15 交换，4 与 7 交换，满足希尔排序思想，所以选 C。
9. C 【详解】比较次数与初态有关的是插入排序，快速排序，所以选 C。
10. D 【详解】归并排序和气泡排序是稳定的，所以选 D。

二、填空题：11~15 小题，每小题 2 分，共 10 分。把符合题目要求的答案填写在下划线处。

11. `+a*-bcd ; abc - d*+`

【详解】中缀表达式转换成前缀表达式需要设置两个栈 R、Q，R 用于存储运算符，Q 用于存储转换后的表达式。对中缀表达式从右至左依次扫描，当遇到操作数时入栈 Q，扫描到操作符时，将操作符压入栈中，进栈的原则是保持栈顶操作符的优先级要高于栈中其他操作符的优先级；否则，将栈顶操作符依次出栈并入栈 Q，直到满足要求为止；遇到“(”进栈，当遇到“(”时，退 R 栈入 Q 直到“(”为止。

中缀表达式转换成后缀表达式要对中缀表达式从左至右依次扫描，由于操作数的顺序保持不变，当遇到操作数时直接输出，设立一个栈用以保存操作符，扫描到操作符时，将操作符压入栈中，进栈的原则是保持栈顶操作符的优先级要高于栈中其他操作符的优先级；否则，将栈顶操作符依次退栈并输出，直到满足要求为止；遇到“(”进栈，当遇到“(”时，退栈输出直到“(”为止。

12. `n(n-1) ; n`

【详解】图强连通的定义是任意两点之间都有一条路径，最多时每个顶点发出 $n-1$ 条边，n 个顶点共有 $n(n-1)$ 条边。最少时每个结点至少要一条出路(单节点除外)，至少有 n 条边，正好可以组成一个环。

13. `63 ; 20`

【详解】最多时满二叉树，即 $2^6-1=63$ ，最少时可以利用公式 $N_h=N_{h-1}+N_{h-2}+1$ 求得，其中 N_h 代表高度为 h 的 AVL 树最少结点树， N_{h-1} 代表 AVL 树的一棵子树， N_{h-2} 代表另一棵子树， $N_0=0$ ， $N_1=1$ ， $N_2=2$ ，可以计算出最少为 20。

14. `242 ; 31`

【详解】根据 B-树定义，3 阶 B 树最多 $3^3-1=242$ ，第 h 层最少包含结点数为 $2 \lceil \frac{m}{2} \rceil^{h-1}$ ，所以高度为 5 最少关键字数 $N \geq 2 \lceil \frac{3}{2} \rceil^{5-1}-1=31$

15. `4 ; 1`

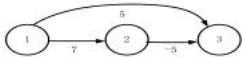
【详解】最多比较是树的高度 $\log_2 11$ ，取整为 4。最少比较 1 次。

三、简答题：16~17 小题，每小题 10 分，共 20 分。



16.

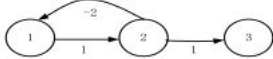
(1) 例如:



以顶点 1 为源点, 按 Dijkstra 算法, 顶点 1 到顶点 3 的最短路径为 (1, 3), 长度为 5; 实际的最短路径为 (1, 2, 3), 长度为 2。(3 分)

(2) 例如, 在公共交通费用网络中, 对于某段路程, 乘公交车单位给补助, 而打车实报实销。如果补助费用大于打车费用, 则这段路程的费用即为负数值。(3 分)

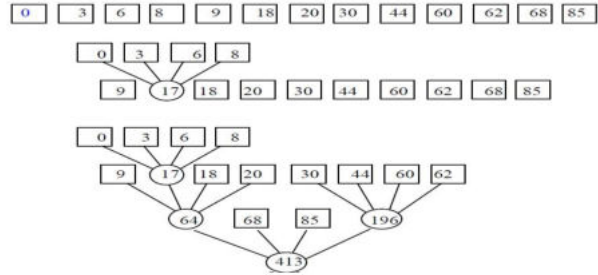
(3) 例如:



顶点 1 到顶点 2 的路径长度可以任意小。(4 分)

17. (1) 初始归并段个数 $n = 12$, 外归并路数 $k = 4$, 计算 $(n-1) \% (k-1) = 11 \% 3 = 2 \neq 0$, 必须补 $k-1-2=1$ 个长度为 0 的空归并段, 才能构造 k 路归并树。此时, 归并树的内结点应有 $(n-1+1)/(k-1) = 12/3 = 4$ 个。(2 分)

最佳归并树图示如下:



(4分)

(2) $WPL = (3+6+8)*3 + (9+18+20+30+44+60+62)*2 + (68+85)*1 = 690$ (2分)

访问外存的次数: $690 \times 2 = 1380$

(2分)

四、算法设计题: 18~19 小题, 每小题 10 分, 共 20 分。按以下要求设计算法: (1)给出算法的基本设计思想; (2)根据设计思想, 采用 C 或 C++或 JAVA 语言描述算法。(3)说明你所设计算法的时间和空间复杂度。

18. 分析: 因为字符只有 9 种可能, 所以可以申请一个 10 大小的数组来统计每个字符出现的次数(以空间换时间), 统计完后, 我们以原数组的元素值为下标, 访问统计数组, 直到遇到第一个访问数组元素值为 1 的元素, 其所在位置的字符即为第一个只出现一次的字符。

(1) 算法的基本设计思想: (2 分)

首先建立一个长度为 10 的计数数组 (Hash 表), 存放每个数字的出现次数, 初始时各单元 (桶) 均为 0;
然后, 扫描字符串, 每碰到一个字符 (数字), 在 Hash 表中找到对应的项并把出现的次数增加一次。
最后, 扫描计数数组 (Hash 表), 得到每个字符出现的次数了。

(2) 算法描述 (6 分)

void FindChar(char* str)

https://blog.csdn.net/weixin_45406155



dream or nightmare

关注

```
{
    const int tableSize = 10;
    unsigned int hashTable[tableSize];
    if(str==NULL)
        return;
    int hashTable [tableSize]={0};
    for(int i=0;i<strlen(str);i++)
        hashTable [str[i]]++;
    for(int j=0;j<strlen(str);j++)
    {
        if(hashTable [str[j]]==1)
        {
            printf("%c\n",str[j]);
            return;
        }
    }
    if(j==strlen(str))
        printf("不存在这样的字符!\n");
}
```

(3) 时间和空间复杂度 (2 分)

算法的基本操作时扫描字符串的每个字符并对其出现次数进行计数，因此时间复杂度为 $\Theta(n)$ ，满足题目要求；

算法的辅助空间开销，取决于字符串对应的字母表的大小，与 n 无关，因此空间复杂度为 $O(1)$ 。

19. (1) 算法的基本设计思想 (2 分)

①如果 AVL 树的左子树的结点数为 $a=k-1$ ，则根结点的关键字即为第 k 小的关键字；否则，转②。

②如果 AVL 树的左子树的结点数 a 大于 $k-1$ ，则递归地其左子树上查找 AVL 树中第 k 小的关键字；否则转③。

③ (AVL 树的左子树的结点数 $a < k-1$)，则递归地其右子树上查找 AVL 树中第 $k-a-1$ 小的关键字，即为 AVL 树中第 k 小的关键字。

(2) 算法描述 (6 分)

根据题目要求，算法的存储结构可定义如下：

```
typedef struct celltype{
    records key;
    int size;
    struct celltype *lchild,*rchild;
}AVLNode;
typedef AVLNode *AVLTree;
算法实现如下：
AVLNode *FindKthKey(AVLTree root, int k)
{
    if(root->size<=k) return NULL;
    if(root->lchild->size==k-1)
        return root;
    else if(root->lchild->size>k-1)
        FindKthKey(root->lchild, k);
    else
        FindKthKey(root->rchild, k- root->lchild->size-1);
}
```

(3) 时间和空间复杂度 (2 分)

算法的本质就是递归地查找 (Search) AVL 树，因此时间和空间性能都取决于 AVL 树的高度，因此，时间复杂度为 $\Theta(\log n)$ ，满足题目要求；空间复杂度亦为 $\Theta(\log n)$ 。

https://blog.csdn.net/weixin_45406155

其他年份的题

- 哈工大2012秋数据结构期末试题(含答案)
- 哈工大2019秋数据结构期末试题
- 哈工大2015秋数据结构期末试题(含答案)
- 哈工大2017秋数据结构期末试题(含答案)
- 找到了其他年份的接着更新

(哈工大2016年数据结构期末题及答案由于版权原因不能发到博客上，需要的私聊我)
需要pdf的可以联系我

文章知识点与官方知识档案匹配，可进一步学习相关知识

算法技能树 首页 概览 55287 人正在系统学习中

「数据安全 & 隐私计算」全系列免费公开课！产学研40+豪华嘉宾阵容，共同打造全景知识地图

作为数据要素流通的实践探索者与隐私计算技术的布道者，隐语社区联合学术届与产业界的数十位专家学者，于2022年11月正式推出「隐私计算技术线上MOOC」第一期

2 条评论 qq_44125033 热评 选择题第二个，先排序在插入的话logn不就可以了么

 dream or nightmare

关注