

清华大学2001年数据结构试题 PDF转换可能丢失图片或格式
，建议阅读原文

https://www.100test.com/kao_ti2020/112/2021_2022__E6_B8_85_E5_8D_8E_E5_A4_A7_E5_c73_112461.htm 试题内容: 一、试给出

下列有关并查集(mfsets)的操作序列的运算结果: union(1,2), union(3,4), union(3,5), union(1,7), union(3,6), union(8,9), union(1,8), union(3,10), union(3,11), union(3,12), union(3,13), union(14,15), union(16,0), union(14,16), union(1,3), union(1,14)。(union是合并运算,在以前的书中命名为merge) 要求 (1) 对于union(i,j),以i作为j的双亲; (5分) (2) 按i和j为根的树的高度实现union(i,j),高度大者为高度小者的双亲; (5分) (3) 按i和j为根的树的结点数实现union(i,j),结点数大者为结点数小者的双亲; (5分) 二、设在4地(A,B,C,D)之间架设有6座桥,如图所示: 要求从某一地出发,经过每座桥恰巧一次,最后仍回到原地 (1) 试就以上图形说明:此问题有解的条件是什么? (5分) (2) 设图中的顶点数为n,试用C或Pascal描述与求解此问题有关的数据结构并编写一个算法,找出满足要求的一条回路。(10分) 三、针对以下情况确定非递归的归并排序的运行时间(数据比较次数与移动次数): (1) 输入的n个数据全部有序。(5分) (2) 输入的n个数据全部逆向有序。(5分) (3) 随机地输入n个数据。(5分) 四、简单回答有关AVL树的问题。(1) 在有N个结点的AVL树中,为结点增加一个存放结点高度的数据成员,那么每一个结点需要增加多少个字位(bit)? (5分) (2) 若每一个结点中的高度计数器有8bit,那么这样的AVL树可以有多少层?最少有多少个关键码? (5分) 五、设一个散列表包含hashSize=13个表项,其下标从0到12,采用线性探查法解决冲

突. 请按以下要求,将下列关键码散列到表中. 10 100 32 45 58 126
3 29 200 400 0 (1) 散列函数采用除留余数法,用%hashSize(取余运算)将各关键码映像到表中. 请指出每一个产生冲突的关键码可能产生多少次冲突. (7分) (2) 散列函数采用先将关键码各位数字折叠相加,再用%hashSize将相加的结果映像到表中的办法. 请指出每一个产生冲突的关键码可能产生多少次冲突. (8分)

六、设一棵二叉树的结点定义为 struct BinTreeNode{ ElemType data. BinTreeNode *leftChild, *rightChild. } 现采用输入广义表表示建立二叉树. 具体规定如下: (1) 树的根结点作为由子树构成的表的表名放在表的最前面. (2) 每个结点的左子树和右子树用逗号隔开. 若仅有右子树没有左子树,逗号不能省略. (3) 在整个广义表表示输入的结尾加上一个特殊的符号(例如" # ")表示输入结果. 例如,对于如右图所示的二叉树,其广义表表示为A(B(D,E(G,)),C,(F)) A /\ B C /\ D E F / G 此算法的基本思路是:依次从保存广义表的字符串Is中输入每个字符. 若遇到的是字母(假定以字母作为结点的值),则表示是结点的值,应为它建立一个新的结点,并把该结点作为左子女(当k=1)或有子女(当k=2)链接到其双亲结点上. 若遇到的是左括号" (" ,则表明子表的开始,将k置为1.若遇到的是右括号")" ,则表明子表结果. 若遇到的是逗号" ," ,则表示以左子女为根的子树处理完毕,应接着处理以右子女为根的子树,将k置为2. 在算法中使用了一个栈s,在进入子表之前,将根结点指针进栈,以便括号内的子女链接之用. 在子表处理结束时退栈. 相关的栈操作如下: MakeEmpty(s) 置空栈 Push(s,p) 元素p进栈 Pop(s) 进栈 Top(s) 存取栈顶元素的函数 下面给出了建立二叉树的算法,其中有5个语句缺失. 请阅读此算法并把缺失的语句补上. (每

空3分) Void CreateBinTree(BinTreeNode *amp.#.,int left,int right){ Type temp. If(leftType pivot=median3(a,left,right). Int l=left, j=right-1. For(. .){ While(iWhile(iif(itemp=a[i]. a[j]=a[i]. a[i]=temp. l . j--. } else break. } if(a[i]>pivot) {temp=a[i]: a[i]=a[right]. a[right]=temp.} quicksort(a,left,i-1). //递归排序左子区间 quicksort(a,i 1,right). //递归排序右子区间 } } (1) 用C或Pascal实现三者取中子程序 median3(a,left,right). (5分) (2) 改写 quicksort 算法, 不用栈消去第二个递归调用 quicksort(a,i 1,right). (5分) (3) 继续改写 quicksort 算法, 用栈消去剩下的递归调用. (5分) 100Test 下载频道开通 , 各类考试题目直接下载。 详细请访问 www.100test.com