

2021 年硕士研究生招生考试初试(笔试)试题 (A 卷)

科目代码: 911

科目名称: 数据结构

注意: ①认真阅读答题纸上的注意事项; ②所有答案必须写在答题纸上, 写在本试题纸或草稿纸上均无效; ③本试题纸须随答题纸一起装入试题袋中交回!

一. 名词解释(共 6 小题, 每题 5 分, 共 30 分)

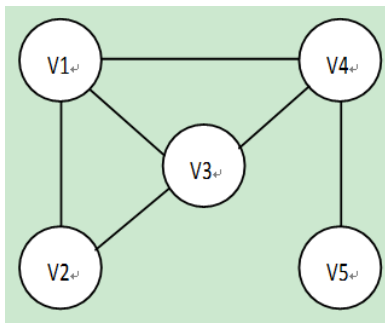
1. 物理结构
2. 数据类型
3. 算法
4. 二叉平衡树
5. 十字链表
6. 关节点

二. 简答题(共 4 小题, 每题 10 分, 共 40 分)

1. 结构是指数据元素之间的相互关系。请简要描述常用基本结构。
2. 请简述希尔排序基本原理。
3. 请简述哈希表的基本工作原理。
4. 一棵二叉树, 叶子结点数为 n_0 , 度为 2 的结点数为 n_2 。证明等式成立: $n_0 = n_2 + 1$

三. 综合应用题(共 8 小题, 每题 10 分, 共 80 分)

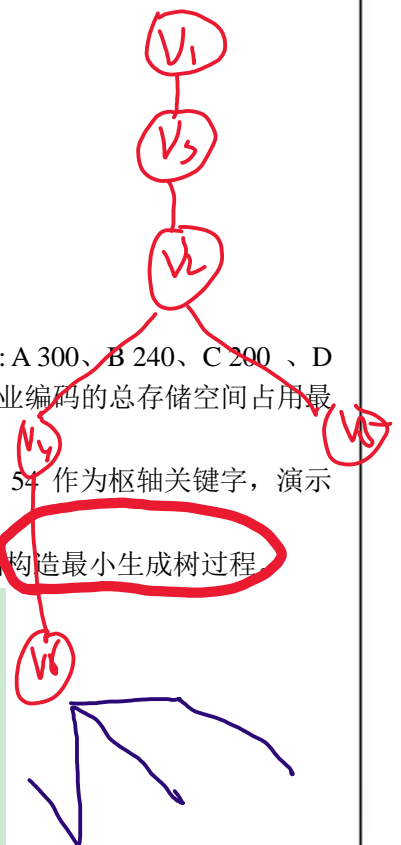
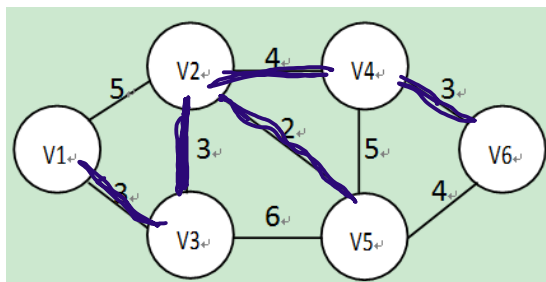
1. 已知二叉树中序遍历序列为 ABCDEFGH; 后序遍历序列为 BADCGHFE. 请写出前序遍历序列。
2. 已知无向图 G, 如图所示。请画出其邻接矩阵, 并写出从 V1 开始深度优先遍历的结点序列。



1. 集合, 线性, 树型, 图形
2. 对待排记录序列先做“宏观”调整, 再作“微观”调整。
3. 将记录的存储位置与它的关键字之间建立一个确定的关系 H, 使每个关键字和唯一的存储位置对应。在查找时, 只需要根据对应关系计算出给定的关键值 $H(K)$, 就可以得到记录的存储位置。

4. 总结点 v 趟边数 e 的交换过程。19 11 31 54 8 78 29 04 64

$v = e + 1, e = n_1 + 2n_2, v = n_0 + n_1 + n_2$



6. 已知有向无环图 AOE 网 G，边代表活动，采用邻接表存储，其拓扑排序后顶点存储于 T 栈，数组 ve[i] 记录每个顶点事件 i 的最早开始时间。请把如下计算关键路径的算法补充完整。

Status CriticalPath (AOEGraph G)

```
{
    vl[0..G.vexnum-1]=ve[0..G.vexnum-1]; //初始化顶点最迟开始时间
    while(!StackEmpty(T))
        for(Pop(T, j)), p=G.vertices[j].firstarc; p; p=p->nextarc)
        {
            k=p->adjvex;
            dut=(p->info); //耗时权重
            if(vl[k]-dut<vl[j]) (1) vl[j]=vl[k]-dut;
        }

    for(j=0; j<G.vexnum; ++j)
    {
        for(p=G.vertices[j].firstarc; p; p=p->nextarc)
        {
            k=p->adjvex;
            dut=(p->info);
            (2) ee=ve[j];
            (3) el=vl[k]-dut;
            tag=(ee==el)?'*':' ';
            printf( j, k, dut, ee, el, tag); //输出关键活动
        }
    }
}
```

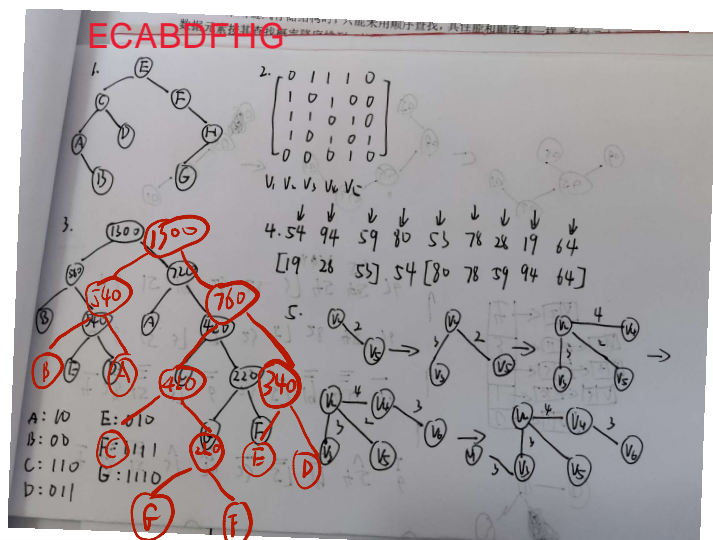
7. 已知 La 和 Lb 是按元素值递增排列的单链线性表。写算法把 La 和 Lb 合并为 Lc，保持数据递增排列规律，并分析其时间复杂度。

```
void Merge_List( LinkList &La, LinkList &Lb, LinkList &Lc)
{
    // ...
}
```

8. 请把计算二叉树深度函数 TreeDepth() 补充完整。

```
typedef struct BinaryTree
{
    int data;
    struct BinaryTree *Left;
    struct BinaryTree *Right;
    struct BinaryTree *Parent;
}Node;

//递归求二叉树深度
int TreeDepth(BinaryTree* root)
{
    // ...
}
```



A : 01
B : 00
C : 100
D : 111
E : 110
F : 1011
G : 1010