

上海大学 2015 ~ 2016 学年 春 季学期试卷

成绩

课程名: 数据结构(二) 课程号: 08305009 学分: 4(A)

应试人声明:

我保证遵守《上海大学学生手册》中的《上海大学考场规则》, 如有考试违纪、作弊行为, 愿意接受《上海大学学生考试违纪、作弊行为界定及处分规定》的纪律处分。

应试人 _____ 应试人学号 _____ 应试人所在院系 _____

题号	一	二	三	四	五	六
	10 分	10 分	10 分	40 分	20 分	10 分
得分						

一、单选题(本大题共 10 小题, 每小题 1 分, 共 10 分)

得分

在每小题列出的备选项中有一个是符合题目要求的, 请将其代码填写在题后的括号内。错选、多选或未选均无分。

- 如果某图的邻接矩阵是对角线元素均为零的上三角矩阵, 则此图一定是 ()。
 - 有向完全图
 - 连通图
 - 强连通图
 - 有向无环图
- 在一个带权连通图 G 中, 权值最小的边一定包含在 G 的 ()。
 - 最小生成树中
 - 深度优先生成树中
 - 广度优先生成树中
 - 关键路径中
- 在排序过程中, 对尚未确定最终位置的所有元素进行一遍处理称为一趟排序。下列排序方法中, 每一趟排序结束都至少能够确定一个元素最终位置的方法是 ()。

I. 简单选择排序 II. 希尔排序 III. 快速排序 IV. 堆排序 V. 二路归并排序

 - 仅 I、III、IV
 - 仅 I、III、V
 - 仅 II、III、IV
 - 仅 III、IV、V
- 将森林 F 转换为对应的二叉树 T , F 中叶结点的个数等于 ()。
 - T 中叶结点的个数
 - T 中度为 1 的结点个数
 - T 中左孩子指针为空的结点个数
 - T 中右孩子指针为空的结点个数
- 给定集合 $s = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$, Union 是合并运算, 经过并查集 (mfsets) 的以下操作: Union(1,2)、Union(7,4)、Union(9,10)、Union(3,2)、Union(1,3)、Union(7,9)、Union(6,5)、Union(6,4)后, 集合 s 分成了 () 个等价类。
 - 2
 - 3
 - 4
 - 5

6. 下列选项中, 不可能是快速排序第 2 趟排序结果的是 ()。

- A. 2,3,5,4,6,7,9 B. 2,7,5,6,4,3,9
C. 3,2,5,4,7,6,9 D. 4,2,3,5,7,6,9

7. 若将关键字 1, 2, 3, 4, 5, 6, 7 依次插入到初始为空的平衡二叉树 T 中, 则 T 中平衡因子为 0 的分支结点的个数是 ()。

- A. 0 B. 1 C. 2 D. 3

8. 对于一组无序的数据, 可以采用 () 查找方法。

- A. 顺序查找 B. 折半查找 C. 插值查找 D. 斐波那契查找

9. 在一个具有 n 个顶点和 e 条边的无向图中, 使用邻接多重表的存储方式, 则边结点个数为 ()。

- A. n B. $2*n$ C. $2e$ D. e

10. 二叉排序树在 () 时其查找效率最高。

- A. 结点太多 B. 完全二叉树 C. 呈单枝树 D. 结点太复杂

二、填空题 (本大题共 10 个空格, 每空格 1 分, 共 10 分)

得分

请在每个空格中填上正确答案。填错、不填均无分。

1. 假定一棵树的广义表表示为 $LS = (A(C, D(E, F, G), H(I, J(B, k))))$, 则树的深度为 _____, 树的度为 _____。

2. 设有向图 G 的二元组形式表示为 $G = (D, R)$, $D = \{1, 2, 3, 4, 5\}$, $R = \{<1, 2>, <2, 4>, <4, 5>, <1, 3>, <3, 2>, <3, 5>\}$, 则给出该图的一种拓扑排序序列 _____。

3. 哈希法存储的基本思想是由 _____ 决定数据的存储地址。

4. 若希望只进行 8 趟排序便能在 4800 个元素中找出其中值最小的 8 个元素, 并且要求排序过程中所进行的关键字比较次数尽可能少, 则应该选用 _____ 排序方法。

5. 当关键字的取值范围是实数集合时, 无法进行 _____ 排序。

6. 若用冒泡排序方法对序列 $\{10, 14, 26, 29, 41, 52\}$ 从大到小排序, 需进行 _____ 次比较。

7. 设有序顺序表中的元素依次为 017, 094, 154, 170, 275, 503, 509, 512, 553, 612。使用折半查找, 则等概率情况下成功的平均查找长度为 _____。

8. 设某无向图中顶点数和边数分别为 n 和 e , 所有顶点的度数之和为 d , 则 $e =$ _____。

9. 在一棵度为 4 的树 T 中, 若有 20 个度为 4 的结点, 10 个度为 3 的结点, 1 个度为 2 的结点, 10 个度为 1 的结点, 则树 T 的叶结点个数是 _____。

三、是非题 (本大题共 10 小题, 每小题 1 分, 共 10 分)

得分

判断下列叙述是否正确, 正确者在括号内打“√”, 错误者在括号内打“×”。

- () 1. 分块查找的基本思想是首先在索引表中进行查找, 以便确定给定的关键字可能存在的块号, 然后再在相应的块内进行顺序查找。
- () 2. 希尔排序的时间复杂度与增量序列的选取有关。
- () 3. 后序遍历一棵平衡二叉树可以得到一个有序的序列。
- () 4. 由同一关键字集合构造的各棵二叉排序树, 其形态不一定相同, 但平均查找长度一定相同。
- () 5. 将树转换为对应的二叉树, 若在二叉树中, 结点 u 是结点 v 的父结点的父结点, 则在原来的树中, u 和 v 具有的关系一定是父子关系。
- () 6. 带权无向图的最小生成树是唯一的。
- () 7. 当向二叉排序树中插入一个结点, 则该结点一定成为叶子结点。
- () 8. 对无向图进行深度优先搜索可以访问到该图中的所有顶点。
- () 9. 直接选择排序在排序过程中, 关键码比较的次数与记录的初始排列顺序无关的。
- () 10. 用邻接矩阵作为图的存储结构时, 则其所占用的存储空间和图中边数有关。

得分

四、应用题 (本大题共 4 小题, 每小题 10 分, 共 40 分)

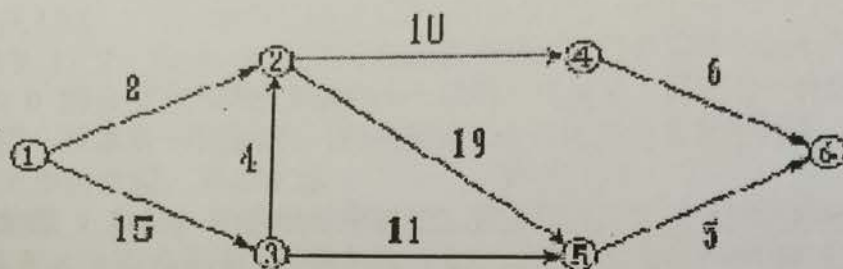
1. 设待排序的排序码序列为{17,16,06,20,18,12,10,28,30,02}, 试分别写出使用快速排序方法, 希尔排序[d={5,3,1}], 基数排序, 堆排序和 2 路归并排序的第二趟排序的结果。可选择的答案如下:

- A: 12,02,06,16,10,17,18,28,30,20
- B: 06,16,17,20,18,12,10,28,30,02
- C: 02,06,16,20,18,12,10,28,30,17
- D: 06,16,17,12,10,18,20,02,28,30
- E: 02,16,06,10,12,17,18,28,30,20
- F: 20,18,12,17,02,06,10,16,28,30
- G: 06,16,10,20,17,12,18,28,30,02
- H: 16,17,06,20,12,18,10,28,02,30
- I: 06,16,17,18,20,10,12,28,02,30
- J: 02,06,10,12,16,17,18,20,28,30

【解答】

排序方法	冒泡排序	快速排序	堆排序	2 路归并排序	基数排序	希尔排序
第二趟排序结果	B					

2. 试对下图所示的 AOE 网络, 解答下列问题。



【解答】

- (1) (2 分) 在拓扑排序序列中一定排在顶点 2 之前的顶点有 _____。
- (2) (4 分) 计算顶点 2 的最早开始时间 $Ve[2]$ 的值为 _____ 和顶点 5 的最晚开始时间 $VI[5]$ 的值为 _____。计算弧 $\langle 2, 5 \rangle$ 的最早开始时间 $e(\langle 2, 5 \rangle)$ 的值为 _____ 和弧 $\langle 2, 5 \rangle$ 的最晚开始时间 $L(\langle 2, 5 \rangle)$ 的值为 _____。
- (3) (2 分) 将该 AOE 网看作无向图, 按克鲁斯卡尔算法求最小生成树时所生成的第二条边为 _____ 而利用普里姆算法从顶点 6 开始求最小生成树所生成的第二条边为 _____。
- (4) (2 分) 将顶点 1 看作单源点, 利用迪杰斯特拉算法求最短路径, 将求得的第二条最短路径的计算过程填入下图。

v	2	3	4	5	6
第一次	2	15	∞	∞	∞
第二次	2				

3. 已知一个森林的先根遍历序列和中根遍历序列如下, 解答问题。

先根: ABECDFGHIJ

中根: EBCDAFHIGJ

1) 构造出这个森林转换的二叉树 (6 分)

【解答】

2) 构造出这个森林。 (4 分)

【解答】

4. 查找算法包括哈希查找, 平衡二叉树查找和 B-树查找, 解答下列问题。

(1) (2 分) 已知一个散列表: 其散列函数为 $h(\text{key}) = \text{key} \% 9$,

试用开放定址法中二次探测再散列解决冲突, 分别在图 1 中插入 42, 33。

下标序号	0	1	2	3	4	5	6	7	8	9
关键字	18				49		24	7	35	

图 1

(2) (2 分) 该散列表在等概率查找时查找成功的平均查找长度为 _____ 以及

该散列表在等概率查找时查找不成功的平均查找长度为 _____。

(共 5 页)

(3) (2分) 若使用平衡二叉树查找方法, 在图 2 建立平衡二叉树过程中依次插入关键字 42, 33.

(4) (4分) 若使用 B-树查找方法, 在图 3 建立 B-树过程中, 依次插入关键字 42, 33, 40 待 B-树建立完毕后, 请删除关键字 7.

【解答】

(3) 画出插入 42 的平衡二叉树

画出在插入 42 之后插入 33 的平衡二叉树

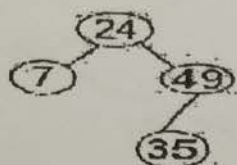
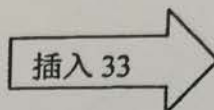
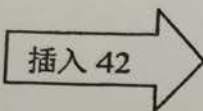


图 2



(4) 依次插入 42, 33, 40 的 B-树建立过程

画出删除关键字 7 的 B-树

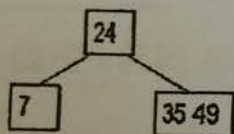


图 3

插入 42

插入 33

插入 40

删除 7

得分

五、算法填空题 (本大题共 2 小题, 每小题 10 分, 共 20 分)

1. 判断给定的二叉树是否为二叉排序树, 补充完整此函数的实现。

```
template<class Type> bool BinarySortTree<Type> :: IsBSTRecurse( )
```

```
{
```

```
    Type pre;    bool result = true;
```

```
    BinTreeNode<Type> *p = (BinTreeNode<Type> *)this->Root( );
```

```
    if (!p)        return true;
```

```
    IsBSTRecurse(p, &pre, &result);
```

```
    return result;
```

```
}
```

```
template<class Type> bool BinarySortTree<Type> :: IsBSTRecurse(BinTreeNode<Type>
*t, Type *last, bool *result)
```

```
{    if ( _____ (1) _____ )
```

```
    {        *result = true;                return *result;    }
```

```
    if (t->GetLeftChild( ) && *result)
```

```
        _____ (2) _____;
```

```
    if ( _____ (3) _____ )    *result = false;
```

```
        _____ (4) _____ ;
```

```
    if (t->GetRightChild( ) && *result)
```

```
        _____ (5) _____;
```

```
    return *result;
```

```
}
```

[解答] (1) _____

(2) _____

(3) _____

(4) _____

(5) _____

(共 5 页)

2. 奇偶交换排序是另一种交换排序。它的第一趟对序列中的所有奇数项 i 扫描, 第二趟对序列中的所有偶数项 i 扫描。若 $A[i] > A[i+1]$, 则交换它们。第三趟有对所有的奇数项, 第四趟对所有的偶数项, ..., 如此反复, 直到整个序列全部排好序为止。补充完整此函数的实现。

```
template<Type> void dataList<Type> :: odd-evenSort ( )
```

```
{
    int i, exchange;
    do {
        (1) _____;
        for ( i=1 ; i < CurrentSize; i += 2 )
            if ( Vector[i] > Vector[i+1] ) {
                (2) _____;
                swap ( Vector[i], Vector[i+1] );
            }
        for ( i=0; i < CurrentSize; i += 2 )
            if ( Vector[i] > Vector[i+1] ) {
                (3) _____;
                (4) _____ ;
            }
    } while ( (5) _____ );
}
```

[解答] (1) _____

(2) _____

(3) _____

(4) _____

(5) _____

六、算法设计题 (10 分)

设计算法判断有向图 G 中 v_1 和 v_2 两个点之间是否存在路径:

编写算法:

```
template <class ElemType> bool pathexist(const AdjListDirGraph<ElemType> &g, int v1,
int v2) {
```