

# 2019 年北京工业大学《数据结构与算法分析》期末考试试卷

## 《数据结构与算法分析》考试试卷 A 卷

考试说明： 考试时间：95 分钟 半开卷

承诺：

本人已学习了《北京工业大学考场规则》和《北京工业大学学生违纪处分条例》，承诺在考试过程中自觉遵守有关规定，服从监考教师管理，诚信考试，做到不违纪、不作弊、不替考。若有违反，愿接受相应的处分。

承诺人： 学号： 班号：

注：本试卷共 四 大题，共 5 页，满分 100 分。

卷面成绩汇总表（阅卷教师填写）

题号	一	二	三	四	总成绩
满分	10	60	15	15	
得分					

★ 请将所有答案写在答题纸上

得分

### 一、填空题（每小题 2 分，共 10 分）

1. 后缀表达式  $21\_15+2*6\_3- / 4+$  的计算结果为（下划线‘\_’表示空格）\_\_\_\_\_。
2. 已知一棵二叉树的前序周游序列是 ABDECFGH，中序周游序列是 DEBAFCHG，该二叉树对应的森林中包含的树的数量是\_\_\_\_\_。
3. 已知一个长度为 13 的顺序表 L，其元素按关键字非递减有序排列，若采用二分检索法在该顺序表 L 中查找一个存在的元素，最多比较的次数为\_\_\_\_\_。
4. 设有向图 G 中有向边的集合  $E=\{<1, 2>, <2, 3>, <1, 4>, <4, 2>, <4, 3>\}$ ，则该图的拓扑序列是\_\_\_\_\_。
5. 广义表  $((a), (b, c), (d, (a, e, f), g), h)$  的长度与深度分别是\_\_\_\_\_。

得分

## 二、简答题（每小题 10 分，共 60 分）

1、一段英文文本“BCDEDABBEDACEEAAEDDEDE”，包含 5 个字符 ‘A’，‘B’，‘C’，‘D’，‘E’，以字符出现的次数作为权值，回答下列问题：

- 画出为这些字符构造的哈夫曼树，构造过程中，约定左子树权值小于右子树权值。
- 给出每个字符各自对应的哈夫曼编码，约定左分支为‘0’，右分支为‘1’。
- 计算带权路径长度 WPL 值。

2、已知关键字序列{ 28, 32, 53, 47, 18, 20, 13, 9, 49 }，请回答如下问题。

- 按照关键字出现次序，画出构造的二叉搜索树，求出其平均检索长度；
- 按照关键字出现次序，画出构造的平衡二叉搜索树，求出其平均检索长度。

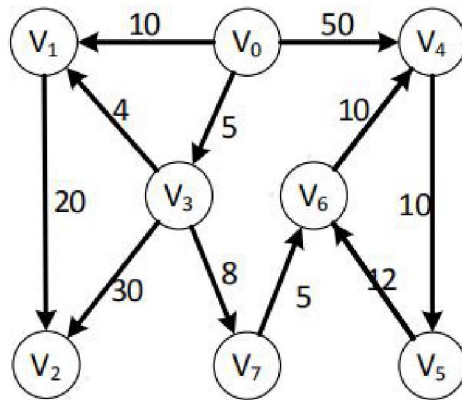
3、已知带权图 G，其相邻矩阵如下所示：

	A	B	C	D	E	F	G
A	0	0	5	3	0	7	9
B	0	0	0	6	2	0	0
C	5	0	0	4	8	0	0
D	3	6	4	0	0	2	0
E	0	2	8	0	0	0	0
F	7	0	0	2	0	0	1
G	9	0	0	0	0	1	0

请回答下列问题：

- 请分别列出该图从顶点 A 开始进行深度优先周游和广度优先周游时得到的顶点序列。
- 按照 Prim 算法，画出以 A 为起点得到的最小生成树。

4、已知有向图 G，如下图所示，填写表格，根据 Dijkstra 算法计算以  $V_0$  为源点到其他各个顶点的最短路径长度的过程。说明：len 当前最短路径长度，pre 为路径序列上到达终点之前最后经过的顶点编号。



迭代步数	S	V <sub>0</sub>	V <sub>1</sub>	V <sub>2</sub>	V <sub>3</sub>	V <sub>4</sub>	V <sub>5</sub>	V <sub>6</sub>	V <sub>7</sub>
初始状态	{V <sub>0</sub> }	len:0 pre:0	len:10 pre:0	len:∞ pre:0	len:5 pre:0	len:50 pre:0	len:∞ pre:0	len:∞ pre:0	len:∞ pre:0
1	{ }	len:0 pre:0	len: pre:	len: pre:	len: pre:	len: pre:	len: pre:	len: pre:	len: pre:
2	{ }	len:0 pre:0	len: pre:	len: pre:	len: pre:	len: pre:	len: pre:	len: pre:	len: pre:
3	{ }	len:0 pre:0	len: pre:	len: pre:	len: pre:	len: pre:	len: pre:	len: pre:	len: pre:
4	{ }	len:0 pre:0	len: pre:	len: pre:	len: pre:	len: pre:	len: pre:	len: pre:	len: pre:
5	{ }	len:0 pre:0	len: pre:	len: pre:	len: pre:	len: pre:	len: pre:	len: pre:	len: pre:
6	{ }	len:0 pre:0	len: pre:	len: pre:	len: pre:	len: pre:	len: pre:	len: pre:	len: pre:
7	{ }	len:0 pre:0	len: pre:	len: pre:	len: pre:	len: pre:	len: pre:	len: pre:	len: pre:

5、已知由9个关键字组成的序列{ 24, 7, 37, 23, 10, 29, 42, 20, 16 }，构造一个长度是13的散列表，散列函数 $\text{Hash}(\text{key}) = \text{key} \% 13$ ，用闭散列线性探测法解决冲突。

(1) 画出构造的散列表。

0	1	2	3	4	5	6	7	8	9	10	11	12

(2) 计算等概率情况下查找成功的平均检索长度 ASL。

(3) 简述删除 24 后（删除位置用墓碑表示），在此表中查找 11 所进行比较的关键字序列。

6、对于关键字序列 (32, 68, 45, 42, 59, 21, 14, 39, 75, 62)，试在答题纸上按照下列格式，写出利用堆排序方法将其重新排列为非递增顺序的前 3 个堆的状态。

初始状态

0	1	2	3	4	5	6	7	8	9
32	68	45	42	59	21	14	39	75	62

第 1 个堆（初始堆）

0	1	2	3	4	5	6	7	8	9

第 2 个堆。

0	1	2	3	4	5	6	7	8	9

第 3 个堆。

0	1	2	3	4	5	6	7	8	9



得分

### 三、算法设计（15 分）

给定一个有向图  $G = (V, E)$ ，以及一个顶点  $V_i$ （顶点编号为  $i$ ），设计算法，求出  $V_i$  的出度和入度的差。要求该算法的设计与存储结构无关，可以既适用于相邻矩阵又适用于邻接表。

图  $G$  的基类声明如下，其中的方法可以在算法实现中直接使用：

```
class Graph{
public:
    int* Mark; //标记图中的顶点是否被访问过
    Graph(int vertex_num); //构造函数，参数为图中顶点数
    virtual ~Graph(); //析构函数
    int GetVerticesNum(); //获得图中顶点的数量
    int GetEdgeNum(); //获得图中边的数量
    void Visit(int vertex); //访问编号为 vertex 的顶点信息
    Edge FirstEdge(int vertex); //获得依附顶点 vertex 的第一条边
    Edge NextEdge(Edge preEdge); //获得与 preEdge 具有相同顶点的下一条边
    bool IsEdge(Edge oneEdge); //判断给定 edge 是否为图中的一条边
    int FromVertex(Edge oneEdge); // 返回 oneEdge 的起始点
    int ToVertex(Edge oneEdge); // 返回 oneEdge 的终点
};

Edge 的声明如下：
class Edge{
public:
    int from, to, weight;
    Edge(); //构造函数
    Edge(int f, int t, int w); //带参数的构造函数
};

算法原型： int DegreeDifferent ( Graph& G, int i ); //i 为顶点编号
```

得分

### 四、数据结构设计（15 分）

教育部对不同的学科按照分类进行不同层次的编号，如：“08”代表“工学”，该类别下的“0812”代表“计算机科学与技术”，继续深入下一层，“081201”代表“计算机系统结构”，“081202”代表“计算机软件与理论”，“081203”代表“计算机应用技术”。在此基础上，可以按照编号进行快速的检索。回答下列问题。

- （1）这些数据采用什么数据结构组织？（文字描述相应的逻辑结构即可）
- （2）写出主要数据结构的抽象数据类型的类定义。