

座位号：

杭州电子科技大学学生考试卷（B）卷

考试课程	数据结构		考试日期	2018 年	月	日	成绩	
课程号	A2701410	教师号		任课教师姓名				
考生姓名		学号（8 位）		年级		专业		

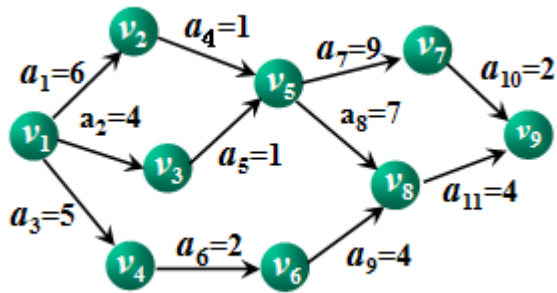
特别提醒：答案一律写在答题纸上，否则不给分。

一. 判断题（每题 2 分，共 10 分）（正确的打“√”，错误的打“×”。）

1. 数据结构中数据对象通常是指性质相同的数据元素的集合。（ ）
2. 线性表采用链式存储结构时，要求内存中可用存储单元的地址是连续的。（ ）
3. 一棵所有结点的度最大为 2 的树是二叉树。（ ）
4. 折半查找(二分查找)要求线性表中的节点必须按关键字递增排列。（ ）
5. 一个有 n ($n>0$) 个顶点的无向图最多有 $n(n-1)$ 条边。（ ）

二. 单选题（每题 2 分，共 30 分）

1. 堆排序的时间复杂度为（ ）。
A. $O(n^2)$ B. $O(n)$
C. $O(n\log n)$ D. $O(n^{3/2})$
2. 链表区别于线性表的特点是（ ）。
A. 可随机访问任一元素
B. 插入、删除不需要移动元素
C. 事先需估计存储空间
D. 所需空间与线性表长度成正比
3. 一个栈的入栈序列为 A、B、C、D、E，则出栈时不可能的输出序列是（ ）。
A. EDCBA B. DECBA C. ABCDE D. DECAB
4. 判断一个顺序队列 sq（容量为 QueueSize）为空队列的条件是（ ）。
A. $sq.rear == sq.front$ B. $sq.rear == 0$
C. $sq.front == QueueSize$ D. $sq.rear == QueueSize + 1$
5. 一个循环队列一旦定义，其占用空间的大小（ ）。
A. 已固定 B. 可以改变
C. 不能固定 D. 动态变化
6. 非空顺序栈 s 出栈(pop)时，用 e 返回其值的语句是（ ）。
A. $e = --*s.top;$ B. $e = *s.top--;$
C. $e = *--s.top;$ D. $e = *(s.top-1);$
7. G 是一个非连通无向图，共有 32 条边，则该图至少有（ ）个顶点。
A. 6 B. 7 C. 8 D. 9
8. 深度为 6 的二叉树，节点数最多可达（ ）个。
A. 32 B. 31 C. 63 D. 64
9. 某项目的 AOE 图如下所示，则该项目完工最少需要（ ）单位时间。
A. 18 B. 17 C. 16 D. 15



10. 广义表 $D = (A, B, C)$ ，其中 A 的长度为 0，B 的长度为 1，C 的长度为 2，则 D 的长度为（ ）。
A. 2 B. 3 C. 4 D. 无法确定
11. 采用顺序查找法查找长度为 n 的线性表时，则表中每个元素的平均查找长度为（ ）。
A. n B. $n/2$ C. $(n-1)/2$ D. $(n+1)/2$
12. 如果图 G 是一个具有 n 个顶点和 e 条边的无向连通图，则必满足（ ）。
A. $n>e$ B. $n<e$ C. $n=e$ D. $e>n-2$
13. 平衡二叉树上所有结点的平衡因子(BF)均不可能为（ ）。
A. -2 B. -1 C. 0 D. 1
14. 对线性表进行二分查找时，要求线性表必须（ ）。
A. 顺序方式存储 B. 链式方式存储
C. 顺序方式存储，且结点按关键字有序排序 D. 链式方式存储，且结点按关键字有序排序
15. 下列排序方法中，所需辅助存储空间最少的是（ ）。
A. 快速排序 B. 堆排序 C. 归并排序 D. 基数排序

三. 填空题（每空 2 分，共 10 分）

1. 表达式 $a+b*c-(d+e)$ 的前缀表达式为_____。
2. 设 S 为一个长度为 n ($n>3$) 的字符串，其中的字符各不相同，则长度为 2 的子串个数为_____。
3. 已知某二叉树的后序遍历次序为 DABEC，中序遍历次序为 DEBAC。其前序遍历次序为_____，层次遍历次序为_____。
4. 在各种内部查找方法中，平均查找长度与结点个数 n 无关的查找方法是_____。

四. 问答题（每题 10 分，共 40 分）

1. 请使用克鲁斯卡尔(Kruskal)算法（5 分）和普里姆(Prim)（5 分）(以 0 点为起点)求出下图的最小生成树,依次写出每次被选择的合法的合并代价最小的边的编号,用一个空格分隔。顶点 a 到顶点 b ($a<b$) 之间的边编号为 ab,例如图中权值为 10 的边编号为 03。

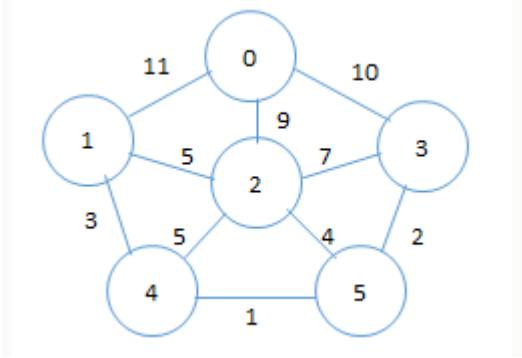


图 1

座位号：

2. 一个包含 100,000 个字符的文件，各字符出现频率如下表所示。

	a	b	c	d	e	f
频率(千次)	45	13	12	16	9	5

- ①、构造赫夫曼（Huffman）树。（7 分）
- ②、确定其对应的赫夫曼编码。（3 分）

3. 设有一组关键字 {19, 01, 23, 14, 55, 20, 84, 27, 68, 11} ，采用哈希函数为
 $H(key) = key \% 13$
采用开放地址法的**二次探测再散列方法**解决冲突，在 0~18 的散列地址空间：

- ①、画出哈希表示意图；（5 分）
- ②、若查找关键字 84，需要依次与哪些关键字比较；（3 分）
- ③、假定每个关键字的查找概率相等，求查找成功时的平均查找长度。（2 分）

4. 以关键字序列（500，87，512，61，907，170，888,275,653,466）为例，分别写出进行以下排列算法进行升序排列时，关键字序列排序的前两趟状态。

- ①起泡排序；（2 分）
- ②直接插入排序；（从 i=2 起作为第一趟排序）（2 分）
- ③简单选择排序；（2 分）
- ④快速排序；（2 分）
- ⑤基数排序。（2 分）

五. 程序设计题（10 分）。

- 1. 当二叉树采用**二叉链表存储结构**存储方式时，写出二叉树及其结点结构的 C 语言实现代码。（4 分）
- 2. 在 1 的基础上，设计一个计算一颗给定二叉树的**所有节点总数**的算法。（6 分）

座位号：

杭州电子科技大学学生考试卷（B）卷答卷

考试课程	数据结构		考试日期	2018 年 月 日		成绩	
课程号	A2701410	教师号		任课教师姓名		王小军、王慧	
考生姓名		学号（8 位）		年级		专业	

一、判断题（每题 2 分，共 10 分）

1. $\sqrt{}$	2. \times	3. \times	4. \times	5. \times
-------------------------	-------------	-------------	-------------	-------------

二、单选题（每选 2 分，共 30 分）

1.C	2.B	3.D	4.A	5.A	6.C	7. D	8.C	9.A	10.B
11.D	12.D	13.A	14.C	15.B					

三、填空题（每空 2 分，共 10 分）

1. $-+a*bc+de$ 或 $+a-*bc+de$	2. $n-1$	3. CEDBA	4. CEDBA	哈希查找
------------------------------	----------	----------	----------	------

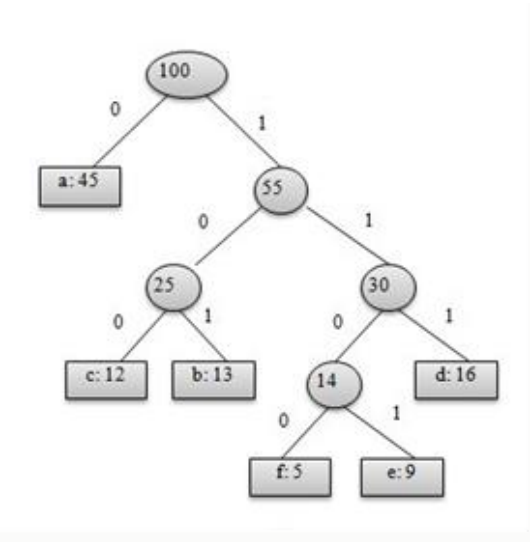
四、问答题（每题 10 分，共 40 分）

1.

Kruskal 算法：45 35 14 25 02 （5 分）（1+ 2+3+4+9=19）

Prim 算法：02 25 45 35 14 （5 分）（9+ 4+1+2+3=19）

2. (1)



(2) a:0 b:101 c:100 d:111 e:1101 f:1100 (每个符号 0.5 分，3 分)

3. (1)

0	1	2	3	4	6	7	8	9	10	11				15	16	17
27	01	14	55	68	84	19	20			23	11					
3	1	2	1	1	3	1	1			1	1					

(2) 关键字：19 20

(3) 平均查找长度：15/10=1.5

4.

(1) 起泡排序排序：

87, 500, 61, 512, 170, 888, 275, 653, 466, 907;

87, 61, 500, 170, 512, 275, 653, 466, 888, 907;

(2) 直接插入排序：

87, 500, 512, 61, 907, 170, 888, 275, 653, 466;

87, 500, 512, 61, 907, 170, 888, 275, 653, 466;

(3) 简单选择排序：

61, 500, 512, 87, 907, 170, 888, 275, 653, 466;

61, 87, 512, 500, 907, 170, 888, 275, 653, 466;

(4) 快速排序：

466, 87, 275, 61, 170, 500, 888, 907, 653, 512;

170, 87, 275, 61, 466, 500, 888, 907, 653, 512;

(5) 基数排序：

500, 170, 61, 512, 653, 275, 466, 87, 907, 888;

500, 907, 512, 653, 61, 466, 170, 275, 87, 888;

五、程序设计题（10 分）。

(1) 参考代码（可以有所不同）

typedef struct node *link;

struct node{Item item; link l,r};

(2) 参考答案（可以有所不同）

```
int nodes( link t){
    int nl,nr;
    if (t==NULL) return 0;
    else if((t->l==NULL)&& (t->r==NULL)) return 1;
    else{
        nl = nodes(t->l);
        nr = nodes(t->r);
        return (nl + nr + 1);
    }
}
```