

2008 级本科班期末考试试卷（A）

《数据结构》参考答案及评分标准

一、选择题（每空 2 分，共 20 分）

1.C 2.A 3.D 4.B 5.A 6.B 7.C 8.C 9.B 10.D

二、简答题（每小题 5 分，共 20 分）

1. 解答：算法的基本特征包括以下几方面：

①有穷性；②确定性；③可行性；④输入；⑤输出

评分标准：每个特性 1 分。

2. 解答：常用的哈希函数构造方法主要有：

① 直接定址法；② 数字分析法；③ 平方取中法；④ 折叠法；

⑤ 除留余数法；⑥ 随机数法；⑦ 乘余取整法等；

评分标准：答出其中的 5 个即可，每个 1 分。

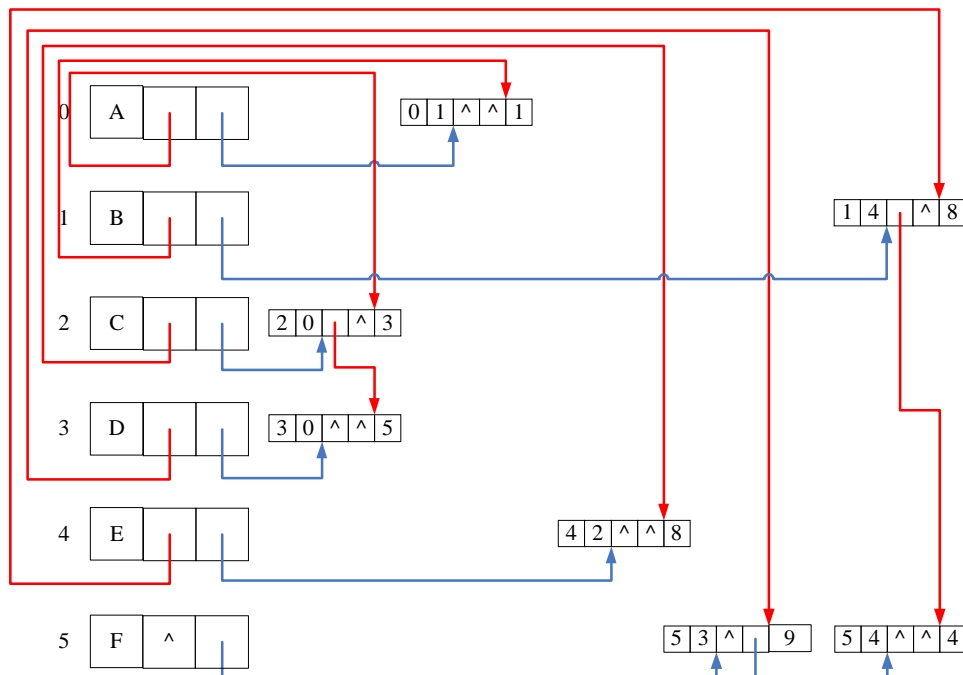
3. 解答：

栈在数据结构的算法中的应用：表达式的计算、二叉树遍历的非递归算法、图的深度优先遍历算法等；

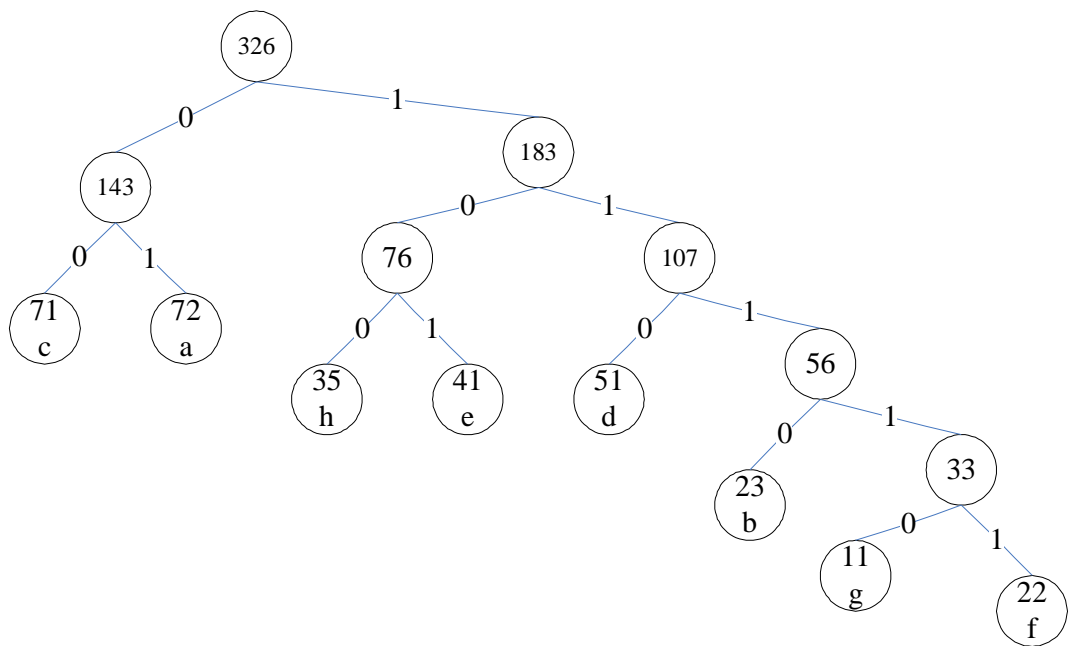
队列在数据结构的算法中的应用：二叉树层次遍历算法、图的广度优先遍历算法、基数排序算法等；

评分标准：各占 2.5 分。

4.



三、（10 分）解答：构造的哈夫曼树如下图所示：



若左支为 0，右支为 1，则各字符的哈夫曼编码为：

**a 01      b 1110      c 00      d 110      e 101      f 11111      g 11110      h 100**

评分标准：此题答案不唯一；哈夫曼树 7 分，哈夫曼编码：3 分。

四、（10 分）解答：图 1 为该森林对应的二叉树。图 2 为森林。

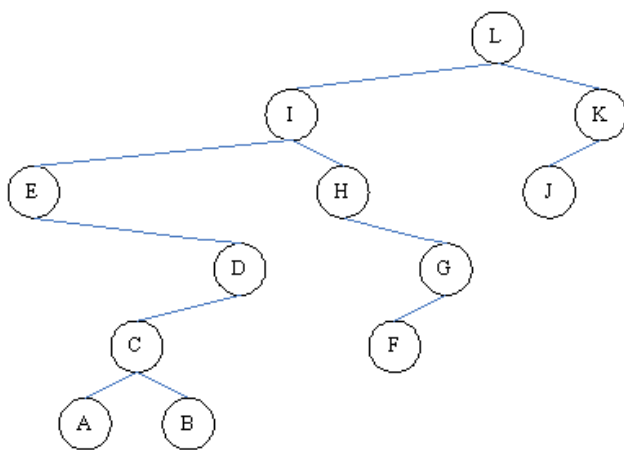


图1 二叉树

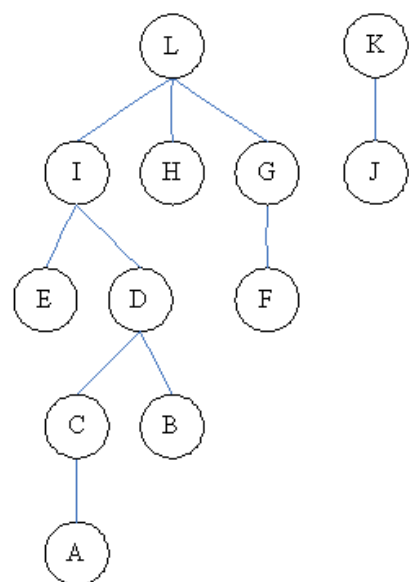


图2 森林

评分标准：二叉树 6 分，森林 4 分。

五、（10 分）

解答：

步骤	顶点 Closededge	2	3	4	5	6	U	V-U	选择
1	adj vex lowcost	1 16		1 21	1 19		{1}	{2, 3, 4, 5, 6}	2 <1, 2>
2	adj vex lowcost		2 5	2 11	1 19	2 6	{1, 2}	{3, 4, 5, 6}	3 <2, 3>
3	adj vex lowcost			2 11	1 19	2 6	{1, 2, 3}	{4, 5, 6}	6 <2, 6>
4	adj vex lowcost			2 11	6 18		{1, 2, 3, 6}	{4, 5}	4 <2, 4>
5	adj vex lowcost				6 18		{1, 2, 3, 6, 4}	{5}	5 <6, 5>

最小生成树的边集合：<1, 2>, <2, 3>, <2, 6>, <2, 4>, <6, 5>

评分标准：结果正确且有过程，得 10 分；

六、（10 分）

解答：

$H(4)=4 \bmod 7=4$

$H(57)=57 \bmod 7=1$

$H(24)=24 \bmod 7=3$

$H(72)=72 \bmod 7=2$

$H(85)=85 \bmod 7=1$ （冲突） $(1+1) \bmod 9=2$

（冲突） $(1-1) \bmod 9=0$

$H(30)=30 \bmod 7=2$ （冲突） $(2+1) \bmod 9=3$

（冲突） $(2-1) \bmod 9=1$

（冲突） $(2+4) \bmod 9=6$

$H(41)=41 \bmod 7=6$ （冲突） $(6+1) \bmod 9=7$

$H(39)=39 \bmod 7=4$ （冲突） $(4+1) \bmod 9=5$

0	1	2	3	4	5	6	7	8
85	57	72	24	4	39	30	41	

$ASL=(1*4+3*1+4*1+2*2)/8=15/8=1.875$

评分标准：哈希表 8 分，ASL 2 分。

七、程序设计题（每题 10 分，共 20 分）

1. 设计递归函数统计二叉树 T 中的结点数据域等于 k 的结点数。

```
int count (TREE *bt)
```

```
{ if (bt==NULL)
```

```
    return 0;
```

```
else
```

```
    return count (bt->lchild)+ count (bt->rchild)+(bt->data==k);
```

```
}
```

评分标准：用到了递归思路，且过程正确得 10 分；没有用到递归扣 5 分。

2. 编写算法利用栈的基本操作实现回文字符串的判断。注：回文串是正读反读都一样的字符串，如"abcba"，"abba"都是回文字符串。

```
int f(char *s)
{ int i,n;
  char ch;
  n=strlen(s); /*求串 s 的长度*/
  initstack(T)
  for(i=0;i<n;i++)
  push(T,s[i]); /*将 s[i]入栈*/
  flag=1;
  for((i=0;(i<n)&&flag;i++))
  { ch=pop(T); /*出栈给 ch*/
    flag=(ch==s[i]); }
  return flag;
}
```

评分标准：用到了栈的思想，且过程正确得 10 分；没有用到栈的思想，得分不能超过 5 分。