

## 2003 年硕士生数据结构招生入学试题 4/2

共四页, 第六页

### 一、填空题 (30 分)

- 1) 选择定义数据结构与数据类型的相关概念。(见 P5, P8)
 

数据结构:  $\phi, d$  ;

数据类型:  $c, d, f$  ;

a) 数据 b) 数据元素 c) 数据对象  
d) 关系 e) 存储结构 f) 基本操作
- 2) 有 K 层的二叉树至多有  $C$  个节点。
 

a)  $2K-1$       b)  $2^{K-1}$       c)  $2^K-1$
- 3) 算法的几个要素为: 有穷性, 确定性, 可行性, 可验证性 (叫停就停)
 

输入, 输出
- 4)  $T(n)$  与  $f(n)$  都用于表述算法的时间复杂性, 简述  $T(n)$  与  $f(n)$  的主要不同之处。
 

$f(n)$  是基本操作重复执行的次数,  $T(n)$  是  
当  $n \rightarrow \infty$  时  $f(n)$  的数量级, 即  $T(n) = O(f(n))$
- 5) 在顺序存储的线性表中插入一个元素, 需要平均移动  $n/2$  个元素, 移动元素的个数与 插入位置 有关。
- 6) 判别以下序列是否为堆。
 

a) (100, 86, 48, 73, 35, 39, 42, 57, 66, 21) 大根堆 ;

b) (12, 70, 33, 65, 24, 56, 48, 92, 86, 33) 不是 ;

c) (103, 97, 56, 38, 66, 23, 42, 12, 30, 52, 6, 20) 大根堆 .
- 7) 已知 L 是无表头结点的循环单链表, 试从下列提供的答案中选择合适的语句序列。
 

a) 在 P 结点后插入 S 结点的语句序列是 (3)(1) ;

b) 在 P 结点前插入 S 结点的语句序列是 (4)(6)(3)(1) .

(1)  $P \rightarrow next = S$  ;

(2)  $P \rightarrow next = S \rightarrow next$  ;

(3)  $S \rightarrow next = P \rightarrow next$  ;

(4)  $Q = P$  ;

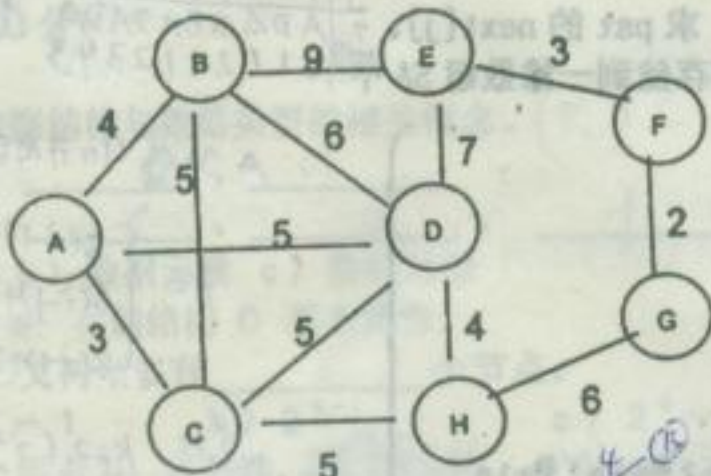
(5)  $P = Q$  ;

(6) while ( $P \rightarrow next \neq Q$ )  $P = P \rightarrow next$  .
- 8) n 个结点的无向完全图, 其边的数目为:  $n(n-1)/2$  ;
- n 个结点的有向完全图, 其边的数目为:  $n(n-1)$  ;
- 9) 简述图与网的区别 网是有向图 .
- 10) 设 A, B, C 三个字符按先后顺序依次进栈且出栈顺序随意; 下面哪个序列为不可能的出栈序列  $C$  .
 

a) ABC      b) ACB      c) BAC

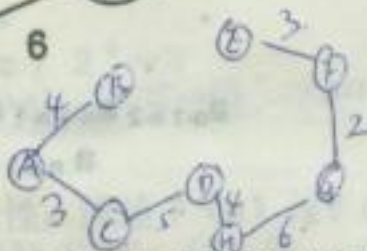


8) 写出下图的邻接矩阵。

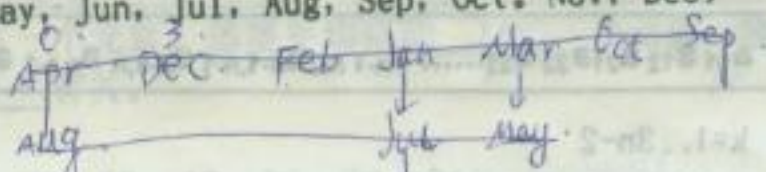
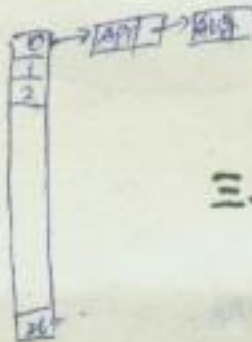


	A	B	C	D	E	F	G	H
A	0	4	3	5	∞	∞	∞	∞
B	4	0	5	6	9	∞	∞	∞
C	3	5	0	5	∞	∞	∞	∞
D	5	6	5	0	7	∞	∞	∞
E	∞	9	∞	7	0	3	∞	∞
F	∞	∞	∞	∞	3	0	2	∞
G	∞	∞	∞	∞	∞	2	0	6
H	∞	∞	5	4	∞	∞	6	0

9) 按普里姆算法求上图的最小生成树。



10) 在地址空间为 0 到 25 的散列区中, 用链地址法构造以下序列的哈希表。  
(Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec)



三、算法阅读与分析 (40分)

1) 算法分析

a) 假设  $n$  为 2 的乘幂, 且  $n > 2$ , 指出下面算法的时间复杂度及变量  $count$  的值

```
int Time(int n)
{
    count=0; x=2;
    while (x<n/2)
    {
        x*=2; count++;
    }
    return(count)
} //Time
```

时间复杂度为  $O(\log_2 n)$ ;  $count = \log_2 \frac{n}{2} - 1$

b) 假设初始调用时  $b=1, e=n$ ; 指出算法的时间复杂度。

```
int Binsearch(Sstabel st, int b, int e, Keytype key) { 折半查找
    if (b<=e) {
        m=(b+e)/2;
        if (EQ(key, st.elem[m]) return m;
        if (LT(key, st.elem[m])
            return Binsearch(st, b, m-1, key);
        else return Binsearch(st, m+1, e, key);
    } else return 0
} //Binsearch
```

$O(\log_2 n)$

二、基础题 (30分)

算法设计 (30分)  
\*\*\*\*以下部分可以用 c、pascal、类c、类pascal描述算法\*\*\*\*

已知线性表中的元素以递增有序排列，并以单链表 L 作存储结构。试写一算法，删除表中所有值大于 min 且小于 max 的元素。(假设数据元素为整型  
结点结构为：(val, next); min, max 作为参数)

试写一个判别表达式中左、右圆括号是否配对出现的算法。(假设表达式为一字符串；算法要求使用栈)

编写一递归算法，计算二叉树中度为 1 的结点数目。

```
1) del(List &L, int min, int max) { (3)
```

// 假设单链表 L 有头结点。

```
s = L; p = L->next;
while (p) {
  if (p->val > min && p->val < max) {
    s->next = p->next;
    p = p->next;
  }
  s = p;
  p = p->next;
}
}
```

```
2) judge (String s) {
```

```
p = s;
while (*p) {
  if (*p == '(')
    push(s, *p);
  else if (*p == ')')
    pop(s);
  p++;
}
```

