

05.5.7 王春己录入

已对 3.9

2004 年上海理工大学硕士研究生入学考试题

考试科目: 数据结构与算法语言 准考证号: _____ 得分: _____

一、填空题 (每小题 2 分, 共 20 分)

1. 某二叉树的前序遍历结点访问顺序是 $abdgcfeh$, 中序遍历的结点访问顺序是 $dgbacchf$, 则其后序遍历的结点访问顺序是 _____。
2. 有一个有序表为 $(1, 3, 9, 12, 32, 41, 45, 62, 75, 77, 82, 95, 100)$, 当二分查找值为 82 的结点时, 进行 _____ 次比较后查找成功。
3. 设有 1000 个无序的元素, 希望用最快的速度挑选出其中前 10 个最大的元素, 最好选用 _____ 排序法。
4. 分别采用堆排序、快速排序、插入排序和归并排序算法对初始状态为递增序列的表按递增顺序排序, 最省时间的是 _____ 算法, 最费时间的是 _____ 算法。
5. 对于线性表 $(18, 25, 63, 50, 42, 32, 90, 66)$ 进行散列存储时, 若选用哈希函数为 $H(k)=k\%9$, (其中 k 为元素值) 则散列地址为 0 的元素有 _____ 个。
6. 具有 m 个叶结点的哈夫曼树共有 _____ 个结点。
7. 设二维数组 $a[10][10]$ 是对称阵, 现将 a 中的上三角 (含对角线) 元素以行为主序存储在首地址为 2000 的存储区域中, 每个元素占 3 个单元, 则元素 $a[6][7]$ 的地址为 _____。
8. 有一棵 50 个结点的完全二叉树, 其叶结点有 _____ 个。
9. m 阶 B 树中除根结点外的所有结点中的关键字个数必须大于等于 _____。
10. 将两个各有 n 个元素的有序表归并成一个有序表, 其最少的比较次数是 _____。

二、设计题 (共 60 分)

1. 已知 la 是带头单链表的头指针, 试编写逆序输出表中各元素的递归算法。(15 分)

2. 设二叉树 T 采用如下存储结构

```
typedef struct tnode
{ TElemType data;
  struct tnode *lchild, *rchild, *parent;
} PBinTree;
```

其中结点的 lchild 域和 rchild 域已分别指向其左、右孩子结点, 而 parent 域中的值为空指针。编写一个递归算法, 将该存储结构中各结点的 parent 域的值修改成指向其双亲结点的指针。(15 分)

3. 编写程序, 求以下级数和的近似值, 约定求和的精度为 0.000001。(15 分)

$$s(x) = x - \frac{x^3}{3 \times 1!} + \frac{x^5}{5 \times 2!} - \frac{x^7}{7 \times 3!} + \Lambda$$

4. 以带头单循环链表作为队列, 只设尾指针 rear, 编写出队函数。(15 分)

三、读程填充 (每个空格 3 分, 共 57 分)

1. 本程序是判一个数是否为回文数。

```
void func(int n, int *y)
{ int m, temp1=0, temp2;
  m=n;
  while ( _____ A _____ )
  { temp2=n%10;
    temp1= _____ B _____ ;
    n = _____ C _____ ;
  }
  if ( m==temp1)*y=1; else *y=0;
}

main()
{ int n, _____ D _____ ;
  scanf("%d", &n);
  func( _____ E _____ );
  if (is==1) printf("YES");
}
```



```

        else printf("NO");
    }

```

2. 本算法是在 s 串中找子串 t。若找到, 则返回子串 t 第一次出现在主串 s 中的位置, 否则返回-1。

```

int index (char s[ ],char t[ ])
{int i=0,j=0;
  while ( _____ A _____ )
    if (s[i+j]==t[j]) j++;
    else { _____ B _____ ; _____ C _____ ; }
    if ( _____ D _____ ) return i;
    else return -1;
}

```

3. 已知无头单链表A和B表示两个集合, 本算法实现 $A=A-B$ (集合的补运算)。

```

typedef struct node {
    int data;
    struct node *next;} lnode;
lnode *setminus(lnode *A, lnode *B)
{lnode *p,*q;
  p= _____ A _____;
  p->next=A; A=p;
  while (B!=NULL)
  {p=A;
   while ( _____ B _____ )
     if(B->data==p->next->data)
       {q=p->next; _____ C _____ ; free(q);}
     else _____ D _____ ;
     _____ E _____ ;
   }
  A=A->next;
  return A;
}

```

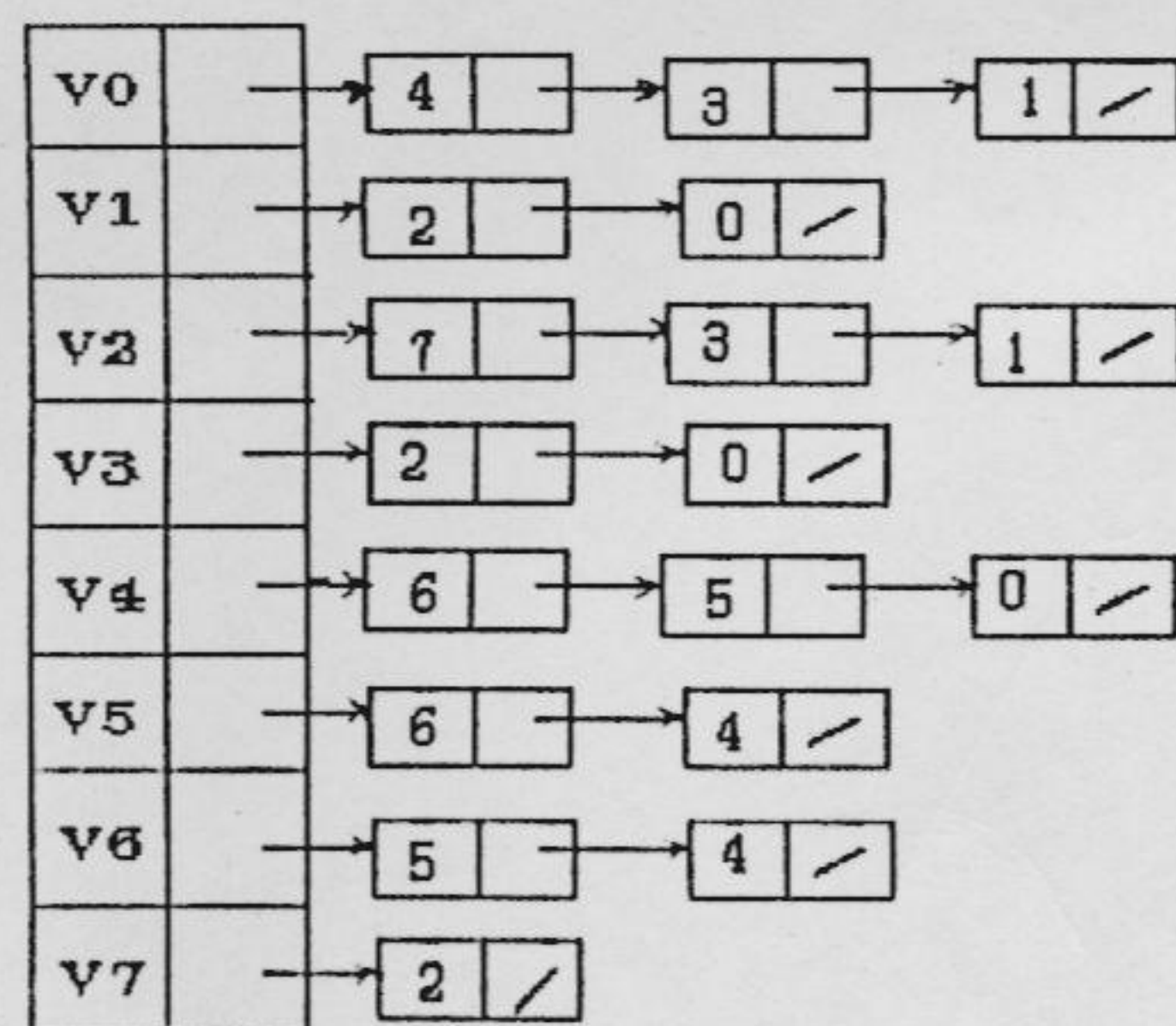

4. 本算法是在二叉树中查找元素值为 word 的结点, 若找到, 则输出从根到该结点之间的路径, 否则输出 "No find".

```
#define Max 100
typedef char datatype;
typedef struct node{
    datatype data;
    struct node *lchild,*rchild;
}bintnode, *bintree;
typedef struct
{
    bintree addr;
    char tag;
}stack;

void path(bintree t,char word)
{
    bintree p;stack s[Max];int find=0,top=0,i;
    p=t;
    do{
        while (p)
            {s[top].addr=p;s[top++].tag='L';
              _____ A _____;}
        while ( _____ B _____ )
            {p=s[top-1].addr;
             if(p->data==word)
                 {for(i=0;i<top;i++)printf("%c",s[i].addr->data);
                  find=1;}
             else top--;
            }
        if ( _____ C _____ )
            { _____ D _____ ; s[top-1].tag='R';}
    }while ( _____ E _____ );
    if(!find)printf("No find");
}
```


四、简答题（共 13 分）

1. 从空树开始，依次插入 20, 30, 50, 55, 60, 52。画出所建平衡二叉树的过程。（5 分）
2. 已知无向图 G 的邻接表，根据如下邻接表画出从 V0 出发的 DFS（深度优先遍历）生成树和 BFS（广度优先遍历）生成树。（8 分）



043LD1A

5