

此试卷资源是由桂电学习资料共享库收集并整理的
桂电学习资料共享库官方 qq 群群号：1014710322



一、应用题（共 50 分）

1.（本题 10 分）已知某二叉树的中序和后序遍历序列分别为 BFDJGACHKE 和 FIGDBKHIECA。

（1）请画出该二叉树。（6 分）

（2）将此二叉树转换成树林。（4 分）

2.（本题 10 分）在数据通信中，需要将传送的文字转换成二进制的字符串，用 0，1 码的不同排列来表示字符。假设需要传送的报文为"AFTER DATA EARED AREARTAREART"，用到的字符集为"A，E，R，T，F，D"（空格字符不计），请为这些字母设计哈夫曼编码。要求：先统计各字符的出现频率，构

建哈夫曼树并编码。在构建哈夫曼树的过程中，遵循左三树根结点权值（小于等于）右子树根结点的规则。

3.（本题 10 分）设某字典组成如下： $D=\{15, 20, 21, 24, 25, 26, 30, 31, 38\}$

（1）若使用散列表对字典进行存储，负载因子 $\alpha=0.75$ ，若以小于散列表长度的最大素数为基础采用除余法构造散列函数，请给出散列函数。画出使用线性探查法处理冲突时装填数据后的散列表。（8 分）

（2）给出查找成功时的平均查找长度。（2 分）

4.（本题 10 分）已知一个带权图的顶点集 $V=\{v_0, v_1, v_2, v_3, v_4, v_5\}$ 。该图的邻接矩阵如图 1 用 Dijkstra 算法求顶点 v_2 到其余各顶点的最短路径。（要求写出解题过程）

	v_0	v_1	v_2	v_3	v_4	v_5
v_0	0	13	8	7	2	∞
v_1	13	0	12	∞	1	∞
v_2	8	12	0	14	∞	∞
v_3	7	∞	14	0	5	3
v_4	2	1	∞	5	0	7
v_5	∞	∞	∞	3	7	0

图 1 邻接矩阵
A-1

5. (本题 10 分)用 P 表示入栈操作 U 代表出栈操作;

(1) 假设栈初始状态为空,若对输入序列 a,b,c,d,e 进行一系列栈操作 SPPUPUPPUUU,得到的输出序列是什么?(3 分)

(2) 假设栈 S 和队列 Q 初始状态为空,元素 1, 2, 3, 4, 5, 6 依次通过栈 S, 一个元素出栈后立即进入队列 Q,若 6 个元素得出队列顺序是 2, 4, 3, 6, 5, 1, 相应得出入栈用 PU 操作串表示是怎么样得? (3 分)

(3) 若用一维数组 $A[0..N-1]$ 存储栈, 令 $A[N-1]$ 为栈底, 用整型变量 i 指示当前栈顶位置, $A[i]$ 是栈顶元素, 则初始化一个空栈时, i 初始值是什么? 当栈中弹出一个数据元素时, i 应该如何变化? 判定栈满的条件是什么? (4 分)

二、算法分析及设计题 (共 50 分)

1. (本题 15 分) 有线性表采用带头结点的链式存储结构, 数据结构及函数 fun 定义如下:

```
typedef int DataType;
```

```
struct node{
```

```
    DataType data ;
```

```
    struct node *next ;
```

```
};
```

```
typedef struct node *LinkList;
```

```
void fun(LinkList L,DataType x,DataType y)
```

```
{if (L==NULL) return:
```

```
LinkList p=L->next;
```

```
while (p!=NULL)
```

```
{if(p->data >x)
```

```
p->data = y;D=D->next;}}
```

(1) 请分析说明 fun 函数的功能 (6 分)

(2) 指出该算法的大 O 阶时间复杂度 (2 分)

2. (本题 15 分) 给定两个集合 SA 和 SB, 分别具有 n 和 m 个整型数据元素, 已知 $n > m$, P 是大于 n 的最小素数, 且 SA 中各元素对 P 取余获得的 n 个输出在 $[0...(p-1)]$ 上无冲突的。以下代码用于求集合 SA 和 SB 的交集。

```
#define MAXNUM 100 //假设集合中的元素都不超过 100
```

```
typedef struct Set{
```

```
Int number; //记录集合中的实际元素个数
```

```
Int *element; //使用数组保存集合中各元素
```

```
}Set;
```

```
Set Intersection( Set *SA, Set *SB)//假设 SA 对应的集合中元素个数多于 SB 表示的集合
```

```
{
```

```
Set *result=(Set*) malloc (sizeof(Set));// 用于存放集合 SA 和 SB 的交集
```

```
Result->element=(int*)| _____ ① _____; //分配存放到交集的存储空间
```

```
int pos=0;
```

```
int p=PrimeNumber(SA->number); // PrimeNumber ( int n)函数能得到大于 n 的最小素数 int *hash=(int*) malloc (p*sizeof(int); //建立散列地址空间
```

```
for(int i=0; i<SA->number; i+)
```

```
_____ ② _____; 将 SA 集合的元素放入到散列地址空间
```

```
/* ③-⑤语句功能概述:遍历 SB 集合的元素, 若使用 SB->element[i]元素计算得到散列表中对应散列地址处存放的数据元素与 SB->element[i]相同,则表示该元素
```


属于 SA 与 SB 的交集*/

```
For ( ③ )  
If( ④ )  
    ⑤  
Return result;  
}
```

(1) 在标号为①的地方，下面哪个选项的代码是最合适的?请说明理由。(3 分)

- A. malloc (SA->number * sizeof(int))
- B. malloc(SB->number * sizeof(int))
- C. malloc((SA->number>SB->number)?SA->number:SB->number * sizeof(int));
- D. malloc (MAXNUM * sizeof(int))

(2) 请填写 Intersection 算法中②-⑤的内容，并保证时间复杂度不超过 $O(n+m)$ 。(12 分)

3.(本题 20 分)现有一个输出函数 Show、一个排序函数 pSort 及主函数代码如下。

void Show(int *A, int N)//输出函数，用于输出数组 A 中 N 个数据元素

```
{  
    int k;  
    for(k=0;k<N;k+1)  
        printf("%d ",A[k]);printf("\n ");  
}
```

//排序函数

void pSort (int A[], int N)

//填写此部分代码}

//主函数

```
Int main (void) {  
    Int b []={11,42,12,5,77,16,91,24};  
    pSort(b,8);  
}
```

完成算法 pSort 后运行该程序,得到运行结果如图 2 所示;



若将主程序中数组 b 改为 int b []={45,2,66,8,15,9,33,4};运行程序后结果为



2	45	4	66	8	15	9	33
2	4	45	8	66	9	15	33
2	4	8	45	9	66	15	33
2	4	8	9	45	15	66	33
2	4	8	9	15	45	33	66
2	4	8	9	15	33	45	66
2	4	8	9	15	33	45	66

图3 程序运行截图

请根据程序运行的结果分析排序采用的方法，按照要求完成以下问题：

- (1) 写出排序算法 pSort 的代码（8 分）
- (2) 分析 pSort 算法的时间复杂度，空间复杂度，以及 pSort 排序算法是否稳定（6 分）。
- (3) 请对 pSort 排序算法进行优化。完成优化后的排序算法 pSort_adv(6 分)。

```
{
//填写此部分代码
}
```

