

华东师范大学 2005 年攻读硕士学位研究生入学试题 共 5 页

考试科目: 数据结构(含 C 语言程序设计)

招生专业: 系统分析与集成、计算机系统结构

计算机软件与理论、计算机应用技术

考生注意:

无论以下试题中是否有答题位置, 均应将答案做在考场另发的答题纸上(写明题号)。

第一部分 C 语言程序设计

一、 填空题(每空 2 分, 共 10 分)

1. 按顺序建立一个名为 file.dat 的有 1000 个空记录的随机存取文件。

```
#include <stdio.h>
#define N 1000
struct DataType{ int acctNum;
                  char name[10];
                  float balance;
                };

main()
{ int i;
  struct DataType record={ 0,"",0.0 }; /* 空记录 */
  (1) cfp;
  if(( (2) )==NULL) printf("File could not be opened.\n");
  else { for(i=0;i<N;i++)
        (3) ; /* 向打开文件写一个空记录 */
        (4) ; /* 关闭打开的文件 */
      }
}
```

2. 设字符数组 password 中记录了用户预先设定的密码, 字符数组 str 接受了用户刚输入的密码。要求在下列函数中填写正确的条件表达式, 使 while 语句能在用户输入正确密码时和用户输入 10 次错误密码时结束循环。

```
void sign_on(char str[], char *password)
{ int a=0;
  while( (1) ) { printf("Enter password please: ");
    gets(str); }
}
```

二、 阅读下列程序并写出程序的执行结果（每题 4 分，共 12 分）

```
1. main()
   { int i;
     for(i=1;i<5;i++) printf("%d  %d\n",i,f(i));
   }
   int f(int a)
   { static int x=1;  return(x*=a); }
```

```
2. #include <stdio.h>
   main()
   { int c;
     if((c=getchar())!='0')
     { main();
       printf("%c",c);
     }
   }
```

设输入数据为：This is 4096<CR> 注：<CR>表示 Enter 键。

```
3. main(){ int i,j;
          printf("*****\n");
          for(i=0;i<3;i++)
          {   j=3;
              switch(i)
              {   case 0:while(j){
                  case 1: printf("i=%d, ",i);
                  case 2: printf("j=%d, ",j);j--;
                      default: ;   }
              } printf("----\n");
          }
}
```

三、 阅读下列函数并说出它完成什么功能（共 8 分）

```
1. #include <stdio.h>
   test(unsigned bits)
   { unsigned i,mask=1<<15,count=0;
     for(i=0;i<16;i++,bits<<=1) if((bits&mask)==mask) count++;
     return(count%2?0:1);
   }
```

四、 按要求编写程序（每题 10 分，共 30 分）

1. 定义一个按“选择法”升序排序的递归函数

函数原型: `void sort(int *p, int n)`

参数 `p`: 为要排序的整型数组的首指针

`n`: 为要排序数组的元素个数

2. 仓库里有一堆物品，某人第一天取走物品的一半还多一件，第二天取走剩余物品的一半多一件。以后每天都取走剩余物品的一半多一件。到第 n 天只剩 a 件物品了，问该堆物品原来共有几件。

设最后一天的物品件数为 x_n ，其前一天的物品件数为 x_{n-1}

按“倒推法”写出相应的迭代公式：

按该迭代公式定义一个求物品总件数的函数，该函数应根据给定的参数 n 和 a 返回 `long` 型的物品总件数值：

3. 设有排成一行的正方形积木 n 块，每块积木的背面分别被涂以红、黄、蓝三种颜色，希望仅通过翻看(`inspect`)和交换(`swap`)这两种操作，最终能将这行积木排列成按红、黄、蓝顺序排列的荷兰国旗的颜色。要求定义一个名为 `holland` 函数来模拟这个游戏过程。说明：

(1) 定义一个全局枚举类型 `enum colour{red,yellow,blue}` 表示颜色的值：

(2) 为了模拟翻看(`inspect`)操作，定义一个只有三个元素的枚举型一维数组 `c`，用来存放 `red`、`yellow`、`blue` 三种颜色。积木则用一个一维指针数组 `pc` 来模拟，`pc` 的元素事先随机指向枚举型一维数组 `c` 中的元素，表示随意排列积木。如下所示：

| | | | |
|----------------|------------------|---------------------|-------------------|
| <code>c</code> | <code>red</code> | <code>yellow</code> | <code>blue</code> |
|----------------|------------------|---------------------|-------------------|

| | | | | | | | | |
|-----------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|
| <code>pc</code> | <code>&c[1]</code> | <code>&c[0]</code> | <code>&c[0]</code> | <code>&c[2]</code> | <code>&c[1]</code> | <code>&c[2]</code> | <code>&c[0]</code> | <code>&c[2]</code> |
|-----------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|

翻看操作用 `inspect` 函数模拟，它将接受外界传递来数组 `pc` 的元素，返回 `pc` 所指对象的值：

(3) 游戏用 `holland` 函数模拟，它将接受外界传递来模拟积木的数组 `pc` 以及数组的元素个数 n （即积木的块数），并将数组排列成荷兰国旗的颜色。如下：

| | | | | | | | | |
|-----------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|
| <code>pc</code> | <code>&c[0]</code> | <code>&c[0]</code> | <code>&c[0]</code> | <code>&c[1]</code> | <code>&c[1]</code> | <code>&c[2]</code> | <code>&c[2]</code> | <code>&c[2]</code> |
|-----------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|

(4) 交换操作用 `swap` 函数模拟，它将交换数组 `pc` 的两个元素的值；

(5) 注意：本题要求定义 `holland`、`inspect`、`swap` 三个函数。

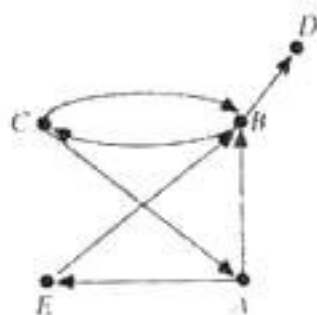
第二部分 数据结构

一、 填空题（每个空格 3 分，共 30 分。）

1. 在长度为 n ($n > 1$) 的顺序表上进行顺序查找，查找不成功的平均查找长度是_____。
2. 设有一个用长度为 n 的数组实现的顺序循环队列，头指针为 f ，指向队列头结点所在的位置，尾指针为 r ，指向队列尾结点后面的一个位置，则队列中的元素个数为_____。
3. 对广义表 $A = ((a, b, c), (d, e, f))$ 做运算 $\text{head}(\text{tail}(\text{head}(\text{tail}(A))))$ ，结果是_____。
4. 数组的顺序存储通常有两种方式：“行序列序”和“列序列序”。对数组 $a[8][18]$ 分别采用这两种方式存储，起始地址均为 1000，数组的每个元素占 5 个单元，则有_____个数组元素在这两种存储方式下的地址是相同的。
5. 已知二叉树有 50 个叶子结点，则该二叉树的总结点数至少有_____个，至多有_____个。
6. 设有一个含 n ($n > 1$) 个结点的大堆（即堆中任意结点的关键字均大于其孩子关键字，若该结点的孩子存在。），该堆存储在一个一维数组中，堆中关键字最小的结点在数组中的下标必定不小于某个整数 x ， x 的最大值是_____。
7. 含 2000 个键值的 5 阶 B 树的高度至少为_____，至多为_____。（注：仅含一个键值的 5 阶 B 树的高度定义为 1。）
8. 对于存储为邻接矩阵的有向图，其边数等于邻接矩阵的_____。

二、 简答题（每小题 6 分，共 30 分）

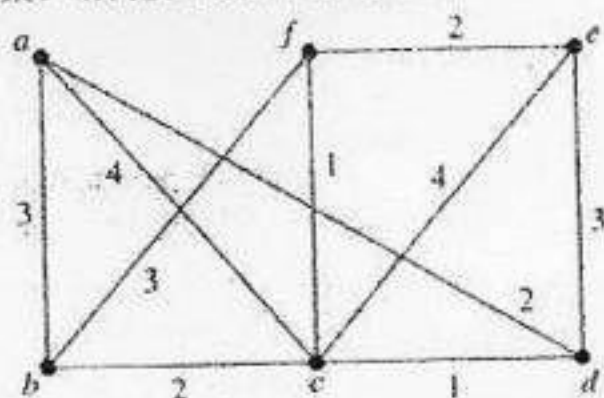
1. 线性链表有那几种常见的变形？请分别简述它们的特点。
2. 给出如下的有向图的邻接表的示意图。



3. 在散列 (hash) 表中，若采用线性探测法（开式寻址法）处理冲突，则同义词在表中总是聚集在一起，占据着连续的区域，上述结论是否正确？为什么？（注：散列表为环形的，即把散列表的第一个位置与最后一个位置看作是相邻的。）

4. 假设有 7 个从小到大排好序的整数的有序表，它们分别含有 10、30、40、50、50、60 和 90 个整数，现要通过 6 次两两合并，将它们最终合并成一个有序表。问：应该按怎样的次序进行这 6 次合并，以使所可能使用的最大的总比较次数最小？请简要地给出求解的过程。

5. 一家石油公司在六个地点有贮油罐 (a、b、d、e 和 f)，现要在这些贮油罐之间建造若干输油管道，以在这些贮油罐之间调配石油，并顺带地向沿途的客户供油。因为建造输油管十分昂贵，所以公司希望建造尽可能少的输油管。另一方面，每条输油管在向客户供油时都会产生些许利润，公司希望所产生的总利润最大。由于各种原因（如地形、距离等等），并非在任意两个贮油罐之间都可以建造输油管，六个贮油罐以及它们之间可以建造的输油管如下图所示，顶点表示贮油罐，边表示可能建造的输油管，边上的权表示相应的输油管所能产生的利润。假设每条输油管的建造费用都相同，请为该公司设计最佳的建造输油管的方案。请简要地给出求解的过程。



三、 算法设计题（每小题 15 分，共 30 分）

- 桶排序也称箱子排序或分配排序，这种排序算法的基本思想是：若待排序的记录的关键字只可能取 m 个不同的值 $k_0 < k_1 < \dots < k_{m-1}$ ，则首先扫描待排序的记录，把关键字等于 k_i 的记录放入（分配到）对应于关键字 k_i 的第 i 个“桶”，然后按 $0, 1, \dots, m-1$ 的次序，输出各个“桶”中的记录，所得到的输出序列即为按关键字从小到大排好序的。
 - 请编写一个对整数顺序表进行桶排序的算法，顺序表中的整数的取值范围是 $0 \sim m-1$ ，其中 m 是某个正整数。能否构造出稳定的桶排序算法？若能构造出稳定的桶排序算法，则使所给出的桶排序算法是稳定的；若不能，则说明理由。
 - 分析上述桶排序算法的时间复杂性和空间复杂性。
 - 通过与其他常见的排序算法作比较，描述桶排序算法的特点，如时空复杂性的特点、优缺点、适用情况等等。
- 请编写一个判断二叉树是否为平衡查找树的算法，要求该算法的时间复杂性最小，且在不考虑系统栈空间的前提下，空间复杂性最小。假定二叉树是以标准形式存储的，树中存储的数据是整数。