

中国科学院软件研究所

一九九八年招收硕士学位研究生入学考试试题

试题名称：软件基础

第一部分：程序设计和数据结构

要求：算法设计题目要求写注解，否则扣分。写出正确的设计思想和伪代码给分。

一. 选择合适的答案填空（1分×20）

- 程序的三种基本控制结构是 _____，_____ 和 _____。
(a) 顺序 (b) 转移 (c) I/O (d) 选择 (e) 重复 (f) 递归
- 过程或函数的副作用是指在过程或函数体内 _____。
(a). 对局部量的修改 (b). 对非局部量的修改
- 在 C 语言的函数 `void f () {static int I=0; ...}` 说明中，静态变量 I 的作用域是 _____，生命期是 _____，I 的初始化是在 _____ 时进行的。（第一、二个空从 (a) ~ (c) 中选择，第三个空从 (d), (e) 中选择）
(a) 整个程序 (b) 函数 f () 所在的文件 (c) 函数 f () 内部
(d) 编译 (e) 程序运行
- 将递归程序转换成非递归程序，一般都要使用栈，但是消除 _____ 可以不使用栈。
(a) 直接递归 (b) 间接递归 (c) 尾递归
- 没有提供指针类型的语言，无法构造链式结构，该说法 _____。
(a) 正确 (b) 错误
- 若广义表 A 满足 $\text{head}(A) = \text{tail}(A) = ()$ ，则 A 为 _____。
(a) () (b) (()) (c) ((), ()) (d) ((), (), ())
- 假定有 K 个关键字互为同义词，若用线性探测法把这 K 个关键字存入散列表中，至少要进行 _____ 次探测。
(a) K-1 次 (b) K 次 (c) K+1 次 (d) $K(K+1)/2$ 次
- 若要尽可能快的完成对实数数组的排序，且要求排序是稳定的，则应选 _____。
(a) 快速排序 (b) 堆排序 (c) 归并排序 (d) 基数排序
- 一个具有 n 个顶点的强连通图，至少有 _____ 条边，至多有 _____ 条边。
(a) n-1 (b) n (c) $n(n-1)$ (d) $n(n-1)/2$
- 用 ISAM 组织文件适合于 _____。
(a) 磁带 (b) 磁盘
- 对外部排序的 K 路平衡归并，采用败者树，归并效率与 K _____。
(a) 有关 (b) 无关
- 若一个具有 n 个顶点，k 条边的无向图是一个森林 ($n > k$)，则该森林中必有 _____ 棵树。
(a) k (b) n (c) n-k (d) 1

13. 已知待排序的 n 个元素可分为 n/k 个组，每个组包含 k 个元素，且任一组内的各元素均分别大于前一组内的所有元素并小于后一组内的所有元素，若采用基于比较的排序，其时间下界应为 _____。
- (a) $O(n \log_2 n)$ (b) $O(n \log_2 k)$ (c) $O(k \log_2 n)$ (d) $O(k \log_2 k)$
14. 若从二叉树的任一结点出发到根的路径上所经过的结点序列按其关键字有序，则该二叉树是 _____。
- (a) 二叉排序树 (b) 哈夫曼树 (c) 堆
15. 用 dfs 遍历一个无环有向图，并在 dfs 算法退栈返回时，打印出相应的顶点，则输出的顶点序列是 _____。
- (a) 逆拓扑有序的 (b) 拓扑有序的 (c) 无序的

二. (共 15 分) 设 $A[0..k-1]$ 是一个整数数组，各分量的初值为 0，即 $A[i]=0$ ($0 \leq i \leq k-1$)，设 n 为连续调用 Demo 过程的次数，请回答下述问题：

- (1) 给出调用次数 $n=4$ 时，数组 A 中各分量的结果。(4 分)
- (2) 若 $n = 2^k$ ，数组 A 中各分量的结果是什么？(4 分)
- (3) Demo 过程的功能是什么？数组 A 起何作用？(4 分)
- (4) 显然 Demo 过程的最坏时间是 $O(k)$ ，故连续调用 Demo 过程 n 次的总时间为 $O(kn)$ 。但是这个界偏大，请详细分析求出一个更小的 n 次连续调用 Demo 的总时间的上界。(3 分)

Procedure Demo (Var A: arraytype; k: integer)

```
{
  i: =0;
  while (i<k) and (A[i]=1) do
  {
    A[i]: =0;
    i: =i+1;
  }
  if i<k then A[i]: =1;
}
```

三. (共 15 分) 一个有向图 $G=(V, E)$ 的转置是有向图 $G^T=(V, E^T)$ ，其中 $E^T = \{ \langle v, u \rangle \in V \times V : \langle u, v \rangle \in E \}$ 。因此将 G 中所有边反向后可得 G^T 。写一算法从给定的 G 求出 G^T ，并分析算法的时间复杂度，要求使用以下邻接表作为存储结构：

```
Type arcptr = ^arcnode;
arcnode = RECORD //边表结点
    arjvex : 1..n; //邻接点序号
    nextarc: arcptr; //指向下一个边表结点
END;
vexnode = RECORD
    vexdata : vextype; //顶点信息
    firstarc : arcptr; //指向边表的第 1 个结点
END;
```

adjlist = ARRAY[1..n] OF vexnode;

第二部分：操作系统和编译原理

四. 简答题 (3 分×5)

- (1) 操作系统的作用是什么？其主要特征是什么？
- (2) 什么是系统调用？它与一般的程序过程调用有什么区别？
- (3) 引起系统抖动的原因是什么？
- (4) 什么是虚拟设备？如何实现虚拟设备？
- (5) UNIX 系统中有哪些磁盘读写方式？

五. (10 分) 分析下面用信号量解决哲学家进餐问题的同步算法是否满足同步机制的四条准则。若不满足，说明为什么，并给出满足同步机制四条准则的同步算法。

```
var fork: array[0..4] of semaphore;
```

```
fork[0] := fork[1] := fork[2] := fork[3] := fork[4] := 1;
```

```
parbegin
```

```
  Pi : repeat /*第 i 个哲学家的生活过程*/
```

```
    ThinkForWhile;
```

```
    P(fork[i]);
```

```
    P(fork[(i+1) mod 5]);
```

```
    EatForWhile;
```

```
    V(fork[i]);
```

```
    V(fork[(i+1) mod 5]);
```

```
  until false
```

```
parend
```

六. (10 分) 下面的二义文法描述命题演算公式，为它写一个等价的非二义文法。

$S \longrightarrow S \text{ and } S \mid S \text{ or } S \mid \text{not } S \mid p \mid q \mid (S)$

七. (15 分) 下面程序的结果是 120。但是如果把第 10 行的 abs (1) 改成 1 的话，则程序结果是 1。试分析为什么会有这样不同的结果。

```
int fact ( )
```

```
{
```

```
  static int i = 5;
```

```
  if (i == 0)
```

```
  {
```

```
    return (1);
```

```
  }
```

```
else
{
    i = i - 1;
    return ((i + abs (1)) * fact ());          /*第 10 行*/
}
}
main ()
{
    printf ("factor of 5 = %d\n", fact ());
}
```

中国科学院软件研究所

一九九八年招收硕士学位研究生入学考试试题答案

试题名称：软件基础

第一部分：程序设计和数据结构

八. 选择合适的答案填空 (1 分×20)

1. (a), (d), (e);
2. (b);
3. (c), (a), (d);
4. (c);
5. (b);
6. (b);
7. (d);
8. (e);
9. (b), (c);
10. (b);
11. (b);
12. (c);
13. (b);
14. (c);
15. (a);

九.

1. 除 $A[2] = 1$ 外, 其余各分量均为 0。
2. A 中分量均为 0。
3. Demo 的功能是二进制数的增量 (加上) 操作, A 相当于一个二进制计数器。
4. n 次连续调用 Demo 的总时间的上界应为 $O(n)$, 详细分析如下:

Demo 操作的时间代价正比于二进制数 A 中位 (每个 $A[i]$ 相当于二进制数的位) 的翻转次数, 而 n 次连续的 Demo 操作是从 0 开始的, 因此在这 n 次增量操作中: $A[0]$ 共翻转 n 次, $A[1]$ 共翻转 $\lfloor n/2 \rfloor$ 次, …… , $A[i]$ 共翻转 $\lfloor n/2^i \rfloor$ 次 ($i \leq \lfloor \log_2 n \rfloor$); 当 $i > \lfloor \log_2 n \rfloor$ 时, 位 $A[i]$ 根本不翻转 (因为数 n 的二进制表示最多有 $\lfloor \log_2 n \rfloor + 1$ 位), 所以 n 次增量操作发生的位翻转总次数为:

$$\sum_{i=0}^{\lfloor \log_2 n \rfloor} \lfloor n/2^i \rfloor < n \sum_{i=0}^{\infty} 1/2^i = 2n = O(n)$$

十.

Procedure transpose (G: adjlist; var GT: adjlist)

```
{
  for i: =1 to n do          //初始化转置图 GT 的邻接表
  {
    GT[i].firstarc := nil;
    GT[i].vexdata := G[i].vexdata;
  }
  for i :=1 to n do
  {
    p := G[i].firstarc;
    while p ≠ nil do
    { //依次搜索 vi 的邻接点 vj
      j := p ↑ .adjvex;    //<vi,vj>是 G 的边
      new(q);              // 申请一表结点
      q ↑ .adjvex := i;
      q ↑ .nextarc := GT[j].firstarc;
      GT[j].firstarc := q;    //将<vj,vi>插入到 GT[j]的邻接表头上
      p := p ↑ .nextarc;    //取 vi 的下一邻接点
    } //end of while
  } // end of for
}
```

时间复杂度: $O(n+e)$, 这里 $n=|V|$, $e=|E|$ 。

第二部分：操作系统和编译

十一.

- (1) 操作系统的作用是：屏蔽系统的硬件特性，提供功能更强的虚拟机；管理计算机系统资源，使之得到有效利用；提供方便友好的用户接口。其主要特征是并发和共享。
- (2) 系统调用是操作系统提供给用户的系统服务程序，是应用程序申请操作系统服务的接口。系统调用运行在系统态，而一般的程序过程运行在用户态；应用程序以直接转入执行的方式调用一般过程，但必须通过中断方式转入执行系统调用，而不能直接转入执行。
- (3) 由于多通道程序系统中引入了过多的程序道数，使得运行进程的大部分时间都用在进行页面的换入换出，而几乎不能完成任何有效的工作。
- (4) 虚拟设备是由脱机或假脱机技术实现的将独占、低速、单台的物理设备改造而成的共享、高速、多台的观念上存在的设备。
- (5) UNIX 系统中有两种读磁盘的方式：一般读方式和提前读方式；三种写方式：一般写方式、异步写方式和延迟写方式。

十二. 不满足“有限等待”准则，当每个哲学家都只拿到一把叉子时，发生死锁。一种改进的算法如下：

```
var fork: array[0..4] of semaphore;
var matex: semaphore;

fork[0] := fork[1] := fork[2] := fork[3] := fork[4] := 1;
matex := 1;
parbegin
  Pi : repeat      //第 i 个哲学家的生活过程
    ThinkForWhile;
    P(matex);
    P(fork[i]);
    P(fork[(i+1) mod 5]);
    V(matex);
    EatForWhile;
    V(fork[i]);
    V(fork[(i+1) mod 5]);
  until false
parend
```

十三.

```
E → E or T | T
T → and F | F
F → not F | p | q | (E)
```

十四.

- (1) i 是静态变量；
- (2) 这和编译时的代码生成策略有关。对于表达式 $(i + \text{abs}(1)) * \text{fact}()$ ，因两个子表达式都有函数调用，因此先产生左子表达式代码，后产生右子表达式代码。当 $\text{abs}(1)$ 改成 1 后，先产生右子表达式代码。因此有不同的结果。