

2002 年攻读硕士学位研究生入学试题

考试科目: 操作系统(含 C 程序设计)

招生专业: 计算机软件与理论、计算机应用技术

操作系统部分 (60 分)

一、解释下列术语(10 分, 每小题 2 分)

- | | |
|----------|-----------|
| 1、分时操作系统 | 2、处理机三级调度 |
| 3、文件目录结构 | 4、临界段 |
| 5、线程 | |

二、填充题 (15 分, 每小格 1 分)

1. 操作系统的结构有_____几种。
2. 操作系统的基本类型有三种, 它们是_____。
3. 进程有三种基本状态, 它们是_____。
4. 进程通信原语有_____。
5. 重定位的方式有_____两种。
6. 主存储器替换策略有_____。
7. 处理机的两种执行模式是_____。
8. 信号量的 wait 和 signal 操作能解决并发进程间的_____问题。
9. 死锁的四个必要条件是_____。
10. 活动头硬盘读写一个扇区总需花 3 种时间是_____。
11. 按控制方式作业可分成_____两种。
12. 操作系统与用户进程间的关系有三种_____。
13. 文件分配的方法有_____。
14. 磁盘调度策略有_____。
15. 实时操作系统的特性是_____。

三、判断改错题(10 分, 每小题 1 分)

判断下列命题是否正确, 若错请改正之, 但有下列划线的关键词不能改。

1. 设计实时操作系统时, 首先要考虑的是如何提高系统的资源的利用率。

2. 多道批处理系统是指那些系统中有多个 CPU 同时处理作业的系统。
3. 进程是一个程序在某数据集上的一次执行, 所以不同进程对应不同的程序。
4. 一个进程中的各线程都有自己的资源, 它们不共享资源。
5. 采用最高优先级调度算法时, 处于运行状态的进程一定是优先级最高的进程。
6. 从一个很大的时间段 (如一天) 上看, 所有的外部设备都是可以为多个作业服务的, 所以不存在独占设备。
7. 作业就是用户提交给系统的程序。
8. 进程在执行中发生缺页中断时, 当操作系统把所缺的页调入主存后, 应该让该进程重新执行被中断的指令。
9. 那些被多个进程共享, 但一次只能被一个进程使用的资源称为临界资源。
10. 银行家算法是一个很好的防止死锁的方法。

四、银行家算法题 (5 分)

若出现下述的资源分配情况:

Process	Current-Allocation	Still-Need	Available
P0	0032	0012	2022
P1	1000	1130	
P2	1301	1310	
P3	0301	0032	
P4	0012	1023	

问: 1、该状态是否安全? 若安全, 则列出一个安全序列。

2、如果进程 P2 提出请求 Request (1、0、1、0) 后, 系统能否将资源分配给它, 写出分析过程。

五、进程调度算法计算题 (5 分)

假定有进程, 它们的提交时间、运行时间如下:

作业号	到达时间	运行时间	开始时间	结束时间
1	0	3		
2	2	4		
3	4	2		
4	6	4		

要求:

- 1、采用先来先服务算法, 分别计算这批进程的平均轮转时间、平均带权轮转时间。
- 2、采用时间片轮转算法 (时间片 $Q=2$), 分别计算这批进程的平均轮转时间、平均带权轮转时间。

六、磁盘调度算法计算题(5分)

若系统中活动头硬盘有 200 个磁道, 现有请求磁盘的磁道号依次为 58, 60, 39, 19, 100, 180, 150, 当前磁头在 50 号磁道上。

要求分别按 FIFO、SSTF、SCAN 算法, 计算它们的平均寻道长度(道数)。

七、请描述利用信号量解决读者——写者问题的程序(读者优先), 并指出什么语句实现进程的读写互斥、写写互斥。(5分)

八、主存替换策略计算题。(5分)

某一进程在页式虚存中的执行程序的页面走向是 2 3 2 1 5 2 4 5 3 2 5 2, 若系统分配给它的帧数为 3 页, 并采用局部淘汰策略。

要求:

分别计算出采用先进先出 FIFO 替换策略、最近最少使用 LRU 替换策略时所缺的页及其次数。

《C 语言程序设计》

一. 阅读程序(本题 10 分)

假定程序 Q 运行时, 从键盘输入到 s 的字符串为:

mail.online.sh.cn

请指出该程序的输出。

[程序 Q]

```
#include <stdio.h>
int ks(char *s, int n)
{ int d=0, c, j;
  if(s[0]!='\0')return(0);
  if(s[0]!='.')d=1;    c=ks(s+1, d==0?d:n+d);
  if(c==0&&d) {
    for(j=0; j<=n; ++j)printf("%c", s[j-n]);    printf("\n");
  }
  return(d==0?d:n+d);
}
main()
{ char s[64];
  printf("Input s: "); gets(s); ks(s, 0);
}
```

二. 按要求写出下列函数定义 (本题 30 分, 每小题 15 分)

1. 写出 group 的函数定义:

函数原型: `int group (NT *t)`

功能说明: 函数 group 按中序扫描二叉树中的所有结点。若某结点 node 有非空的左子树和非空的右子树, 则输出左子树根结点的 v 值、结点 node 的 v 值、右子树根结点的 v 值。

二叉树结点的类型为如下定义的 NT 类型:

```
typedef struct T { char v; struct T *left, *right; } NT;
```

参数说明: t 二叉树根结点的地址。

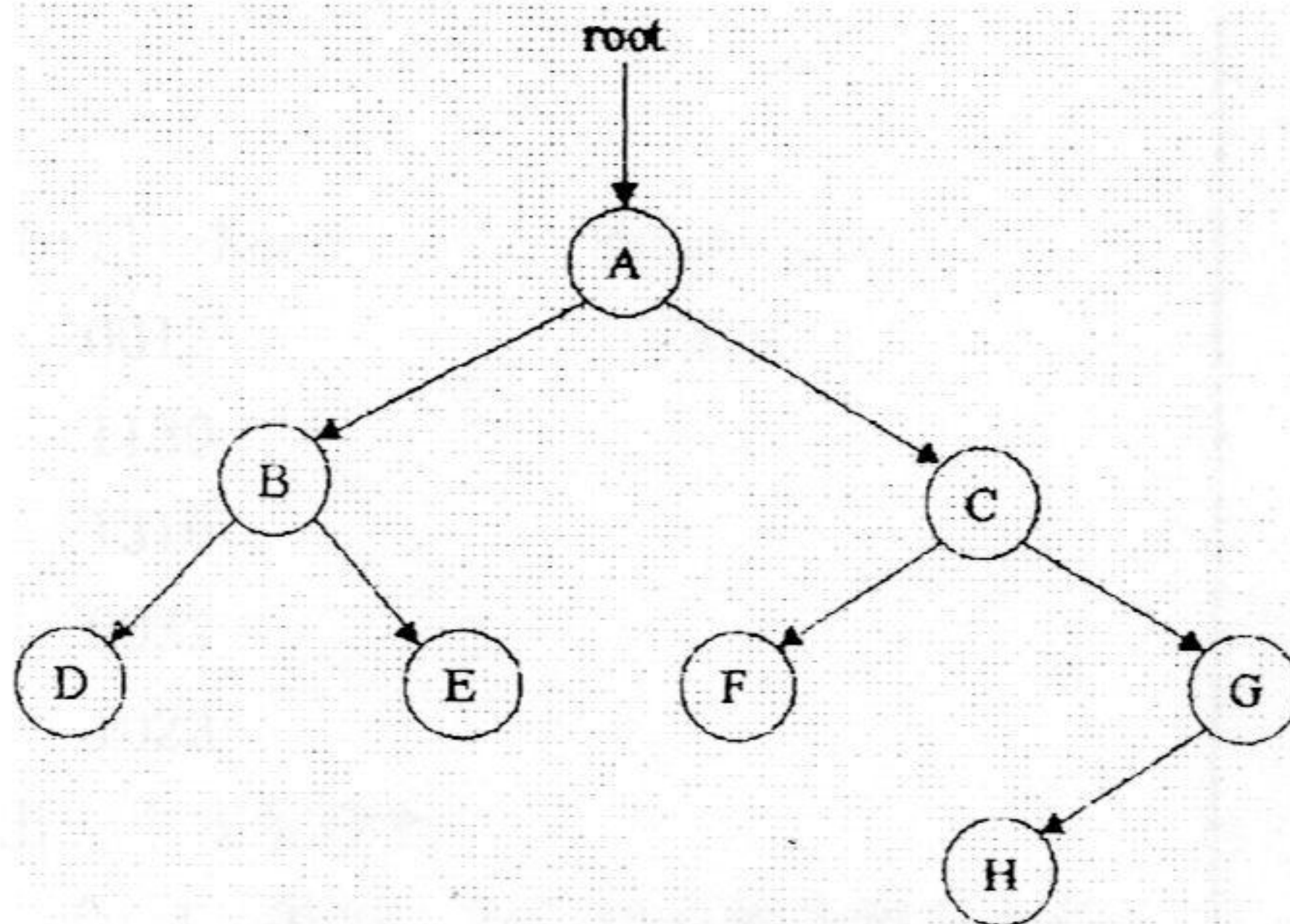
返回值: 左子树、右子树都非空的结点个数。

例如, 若 root 指向如下二叉树的根结点, 则当语句

```
k=group(root);
```

执行完成时, 整数变量 k 获得的函数返回值为 3, 函数 group 产生的输出为:

```
B A C
D B E
F C G
```



2. 写出 convert 的函数定义:

函数原型: `int convert (int d[], int n, int k, char *s, char *sa[])`

功能说明: 函数 convert 把数组 d 中的所有 n 个整数转换成外部形式的 k 进制数 (数字 0 到 k-1 组成的字符串)。

参数说明: d 整数数组首地址。

n 数组 d 中的数据个数。

k 指定的数制。

s 转换后外部形式的 k 进制数 (字符串) 的存贮地址。

sa 各个外部形式的 k 进制数 (字符串) 的首地址。

返回值: 所有 n 个外部形式的 k 进制数 (字符串) 占用的字节数。

例如, 数组 d 中的 3 个数据依次为 15、17、23 时, 语句

```
sn=convert(d, 3, 5, s, sa);
```

执行完成时, 整数变量 sn 得到的函数返回值为 9, 指针数组 sa 和字符串存贮区 s 中的情况如下图所示:

