

## 2016 年计算机学科专业基础综合试题参考答案

### 一、单项选择题

- |       |       |       |       |       |       |       |       |
|-------|-------|-------|-------|-------|-------|-------|-------|
| 1. D  | 2. D  | 3. C  | 4. B  | 5. C  | 6. D  | 7. B  | 8. B  |
| 9. B  | 10. A | 11. D | 12. C | 13. D | 14. A | 15. C | 16. C |
| 17. C | 18. B | 19. B | 20. A | 21. A | 22. A | 23. A | 24. B |
| 25. C | 26. A | 27. B | 28. D | 29. A | 30. C | 31. D | 32. A |
| 33. C | 34. C | 35. D | 36. B | 37. B | 38. D | 39. C | 40. C |

### 二、综合应用题

41. 解析:

(1) TCP 连接的建立分以下三个阶段。首先, H3 向 Web 服务器 S 发出连接请求报文段, 这时首部中的同步位 SYN=1, ACK=0, 同时选择一个初始序号 seq=100。TCP 规定, SYN 报文段 (即 SYN=1 的报文段) 不能携带数据, 但是要消耗一个序号。接着, S 收到连接请求报文段, 为自己选择一个初始序号 seq=y, 向 A 发送确认。这个报文段 SYN=1, ACK=1, seq=y, 确认号 ack 是 100+1=101。它不能携带数据, 但是也要消耗一个序号。最后, H3 收到 S 的确认报文段后, 还要向 S 给出确认。这份确认报文段 SYN=0, ACK=1, 确认号 ack=y+1, 自己的序号 seq=101。因此, 第二次握手 TCP 段的 SYN=1, (1 分) ACK=1; (1 分) 确认序号是 101。(1 分)

(2) 题目规定 S 对收到的每个段 (MSS 大小的段) 进行确认, 并通告新的接收窗口, 而且 TCP 接收缓存仅有数据存入而无数据取出。H3 收到的第 8 个确认段所通告的接收窗口是 20-8=12KB; (1 分) 在慢开始算法里, 发送方 H3 先设置拥塞窗口 cwnd=1KB, 接下来每收到一个对新报文段的确认就使发送方的拥塞窗口加 1KB。H3 共收到 8 个确认段, 所以此时 H3 的拥塞窗口变为 1+8=9KB; (1 分) 发送窗口=min{拥塞窗口, 接收窗口}, 所以 H3 的发送窗口变为 min{9, 12}=9KB。(1 分)

(3) TCP 是用字节作为窗口和序号的单位。当 H3 的发送窗口等于 0KB 时, 也就是接收窗口等于 0KB 时, 下一个待发送段的序号是 20K+101=20×1024+101=20581; (1 分) H3 从发送第 1 个段到发送窗口等于 0KB 时刻为止, 经过五个传输轮次, 每个传输轮次的时间就是往返 RTT, 因此平均数据传输速率是 20KB/(5×200ms)=20KB/s=20.48kbps。(1 分)

(4) 通信结束后, H3 向 S 发送连接释放报文段。S 收到 H3 的连接释放报文段后, 马上发出确认报文段。此时 S 已经没有数据需要传输, 于是它也马上发出连接释放报文段。H3 在收到 S 的连接释放报文段后, 发出确认报文段。S 在收到这份确认后释放 TCP 连接。因此从 t 时刻起, S 释放该连接的最短时间是: H3 的连接释放报文段传送到 S 的时间+S 的连接释放报文段传送到 H3 的时间+H3 的确认报文段传送到 S 的时间=1.5×200ms=300ms。(1 分)

42. 解析:

(1) 根据定义, 正则 k 叉树中仅含有两类结点: 叶结点 (个数记为  $n_0$ ) 和度为 k 的分支结点 (个数记为  $n_1$ )。树 T 中的结点总数  $n=n_0+n_1=n_0+m$ 。树中所含的边数  $e=n-1$ , 这些边均为 m 个度为 k 的结点发出的, 即  $e=m \times k$ 。整理得:  $n_0+m=m \times k+1$ , 故  $n_0=(k-1) \times m+1$ 。(3 分)

(2) 高度为 h 的正则 k 叉树 T 中, 含最多结点的树形为: 除第 h 层外, 第 1 到第 h-1 层的结点都是度为 k 的分支结点; 而第 h 层均为叶结点, 即树是“满”树。此时第 j ( $1 \leq j \leq h$ ) 层结点数为  $k^{j-1}$ , 结点总数  $M_1$  为:

$$M_1 = \sum_{j=1}^h k^{j-1} = \frac{k^h - 1}{k - 1} \quad (3 \text{ 分})$$

含最少结点的正则 k 叉树的树形为: 第 1 层只有根结点, 第 2 到第 h-1 层仅含 1 个分支结点和 k-1 个叶结点, 第 h 层有 k 个叶结点。即除根外第 2 到第 h 层中每层的结点数均为 k, 故 T 中所含结点总数  $M_2$  为:

$$M_2 = 1 + (h-1) \times k \quad (2 \text{ 分})$$

### 【评分说明】

①参考答案仅给出一种推导过程，若考生采用其他推导方法且正确，同样给分。②若考生仅给出结果，但没有推导过程，则（1）、（2）的最高得分分别是 2 分和 3 分。若推导过程或答案不完全正确，酌情给分。

#### 43. 解析：

##### （1）算法的基本设计思想

由题意知，将最小的 $\lfloor n/2 \rfloor$ 个元素放在  $A_1$  中，其余的元素放在  $A_2$  中，分组结果即可满足题目要求。仿照快速排序的思想，基于枢轴将  $n$  个整数划分为两个子集。根据划分后枢轴所处的位置  $i$  分别处理：

①若  $i = \lfloor n/2 \rfloor$ ，则分组完成，算法结束；

②若  $i < \lfloor n/2 \rfloor$ ，则枢轴及之前的所有元素均属于  $A_1$ ，继续对  $i$  之后的元素进行划分；

③若  $i > \lfloor n/2 \rfloor$ ，则枢轴及之后的所有元素均属于  $A_2$ ，继续对  $i$  之前的元素进行划分；

基于该设计思想实现的算法，毋须对全部元素进行全排序，其平均时间复杂度是  $O(n)$ ，空间复杂度是  $O(1)$ 。

##### （2）算法实现（9 分）

```
int setPartition(int a[], int n){
    int pivotkey, low=0, low0=0, high=n-1, high0=n-1, flag=1, k=n/2, i;
    int s1=0, s2=0;
    while(flag) {
        pivotkey=a[low];           //选择枢轴
        while(low<high) {          //基于枢轴对数据进行划分
            while(low<high && a[high]>=pivotkey) --high;
            if(low!=high) a[low]=a[high];
            while(low<high && a[low]<=pivotkey) ++low;
            if(low!=high) a[high]=a[low];
        } //end of while(low<high)
        a[low]=pivotkey;
        if(low==k-1)                //如果枢轴是第 n/2 小元素，划分成功
            flag=0;
        else{                       //是否继续划分
            if(low<k-1){
                low0=++low;
                high=high0;
            }
            else{
                high0=--high;
                low=low0;
            }
        }
    }
    for(i=0;i<k;i++) s1+=a[i];
    for(i=k;i<n;i++) s2+=a[i];
    return s2-s1;
}
```

##### 【（1）（2）的评分说明】

①本题目只需将最大的一半元素与最小的一半元素分组，不需要对所有元素进行全部排序。参考答案基于快速排序思想，采用非递归的方式实现。若考生设计的算法满足题目的功能要求且正确，则（1）、（2）根据所实现算法的平均时间复杂度给分，细则见下表。

时间复杂	分	说明
------	---	----

度	数	
$O(n)$	1 3	采用类似快速排序思想，没有对元素进行全排序。
$O(n\log_2 n)$	1 1	
$O(n^2)$	9	
其他	7	时间复杂度高于 $O(n^2)$ 的算法。

②若在算法的基本设计思想描述中因文字表达没有清晰反映出算法思路，但在算法实现中能够表达出算法思想且正确的，可参照①的标准给分。

③若算法的基本设计思想描述或算法实现中部分正确，可参照①中各种情况的相应给分标准酌情给分。

④参考答案中只给出了使用 C 语言的版本，使用 C++ 语言的答案视同使用 C 语言。

(3) 算法的平均时间复杂度和空间复杂度

本参考答案给出的算法平均时间复杂度是  $O(n)$ ，空间复杂度是  $O(1)$ 。

【评分说明】

44. 解析：

(1) 每传送一个 ASCII 字符，需要传输的位数有 1 位起始位、7 位数据位（ASCII 字符占 7 位）、1 位奇校验位和 1 位停止位，故总位数为  $1+7+1+1=10$ 。（2 分）

I/O 端口每秒钟最多可接收  $1000/0.5=2000$  个字符。（1 分）

【评分说明】对于第一问，若考生回答总位数为 9，则给 1 分。

(2) 一个字符传送时间包括：设备 D 将字符送 I/O 端口的时间、中断响应时间和中断服务程序前 15 条指令的执行时间。时钟周期为  $1/(50\text{MHz})=20\text{ns}$ ，设备 D 将字符送 I/O 端口的时间为  $0.5\text{ms}/20\text{ns}=2.5\times 10^4$  个时钟周期。一个字符的传送时间大约为  $2.5\times 10^4+10+15\times 4=25070$  个时钟周期。完成 1000 个字符传送所需时间大约为  $1000\times 25070=25070000$  个时钟周期。（3 分）

CPU 用于该任务的时间大约为  $1000\times (10+20\times 4)=9\times 10^4$  个时钟周期。（1 分）

在中断响应阶段，CPU 主要进行以下操作：关中断、保护断点和程序状态、识别中断源。（2 分）

【评分说明】

①位于第一问，若答案是 25070020，则同样给分；若答案是 25000000 或 25000020，则给 2 分。如果没有给出分布计算步骤，但算式和结果正确，同样给分。

②对于第三问，只要回答关中断和保护断点，就给 2 分，其他答案酌情给分。

45. 解析：

(1) 页大小为 8KB，页内偏移地址为 13 位，故  $A=B=32-13=19$ ； $D=13$ ； $C=24-13=11$ ；主存块大小为 64B，故  $G=6$ 。2 路组相联，每组数据区容量有  $64\text{B}\times 2=128\text{B}$ ，共有  $64\text{KB}/128\text{B}=512$  组，故  $F=9$ ； $E=24-G-F=24-6-9=9$ 。

因而  $A=19$ ， $B=19$ ， $C=11$ ， $D=13$ ， $E=9$ ， $F=9$ ， $G=6$ 。（各 1 分，共 7 分）

TLB 中标记字段 B 的内容是虚页号，表示该 TLB 项对应哪个虚页的页表项。（1 分）

(2) 块号  $4099=00\ 0001\ 0000\ 0000\ 0011\text{B}$ ，因此，所映射的 Cache 组号为  $0\ 0000\ 0011\text{B}=3$ ，（1 分）对应的 H 字段内容为  $0\ 0000\ 1000\text{B}$ 。（1 分）

(3) Cache 缺失带来的开销小，而处理缺页的开销大。（1 分）因为缺页处理需要访问磁盘，而 Cache 缺失只要访问主存。（1 分）

【评分说明】对于 (3) 中第 2 问，若考生回答因为缺页需要软件实现而 Cache 缺失用硬件实现，则同样给分。

(4) 因为采用直写策略时需要同时写快速存储器和慢速存储器，而写磁盘比写主存慢很多，所以，在 Cache-主存层次，Cache 可以采用直写策略，而在主存-外存（磁盘）层次，修改页面内容时总是采用回写策略。（2 分）

46. 解析：

(1) 由于采用了静态优先数，当就绪队列中总有优先数较小的进程时，优先数较大的进程一直没有机会运行，因而会出现饥饿现象。（2 分）

(2) 优先数 priority 的计算公式为：

$priority = nice + k1 \times cpuTime - k2 \times waitTime$ , 其中  $k1 > 0$ ,  $k2 > 0$ , 用来分别调整  $cpuTime$  和  $waitTime$  在  $priority$  中所占的比例。(3 分)  $waitTime$  可使长时间等待的进程优先数减少, 从而避免出现饥饿现象。(1 分)

【评分说明】

①公式中包含  $nice$  给 1 分, 利用  $cpuTime$  增大优先数给 1 分, 利用  $waitTime$  减少优先数给 1 分; 部分正确, 酌情给分。

②若考生给出包含  $nice$ 、 $cpuTime$  和  $waitTime$  的其他合理的优先数计算方法, 同样给分。

47. 解析:

(1) 两个目录文件  $dir$  和  $dir1$  的内容如下表所示。(3 分)

dir 目录文件		dir1 目录文件	
文件名	簇号	文件名	簇号
dir1	48	file1	100
		file2	200

【评分说明】每个目录项的内容正确给 1 分, 共 3 分。

(2) 由于 FAT 的簇号为 2 个字节, 即 16 比特, 因此在 FAT 表中最多允许  $2^{16}$  (65536) 个表项, 一个 FAT 文件最多包含  $2^{16}$  (65536) 个簇。FAT 的最大长度为  $2^{16} \times 2B = 128KB$ 。(1 分) 文件的最大长度是  $2^{16} \times 4B = 256MB$ 。(1 分)

【评分说明】若考生考虑到文件结束标志、坏块标志等, 且答案正确, 同样给分。

(3) 在 FAT 的每个表项中存放下一个簇号。 $file1$  的簇号 106 存放在 FAT 的 100 号表项中, (1 分) 簇号 108 存放在 FAT 的 106 号表项中。(1 分)

(4) 先在  $dir$  目录文件里找到  $dir1$  的簇号, 然后读取 48 号簇, 得到  $dir1$  目录文件, 接着找到  $file1$  的第一个簇号, 据此在 FAT 里查找  $file1$  的第 5000 个字节所在的簇号, 最后访问磁盘中的该簇。因此, 需要访问目录文件  $dir1$  所在的 48 号簇, (1 分) 及文件  $file1$  的 106 号簇。(1 分)