# 2018年计算机学科专业基础综合试题参考答案

## 一、单项选择题

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

# 二、综合应用题

## 41. 解析:

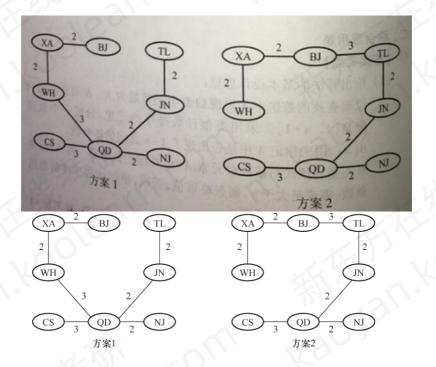
1) 题目要求算法时间上尽可能 效,因此采用空间换时间的办法。分配一个用于标记的数组 B[n],用来记录 A 中是否出现了  $1\sim n$  中的正整数,B[0]对应正整数 1,B[n-1]对应正整数 n,初始化 B 中全部为 0。由于 A 中含有 n 个整数,因此可能返回的值是  $1\sim n+1$ ,当 A 中 n 个数恰好为  $1\sim n$  时返回 n+1。当数组 A 中出现了小于等于 0 或者大于 n 的值时,会导致  $1\sim n$  中出现空余位置,返回结果必然在  $1\sim n$  中,因此对于 A 中出现了小于等于 0 或者大于 n 的值可以不采取任何操作。

经过以上分析可以得出算法流程:从 A[0]开始遍历 A,若 0<A[i]<=n,则令 B[A[i]-1]=1;否则不做操作。对 A 遍历结束后,开始遍历数组 B,若能查找到第一个满足 B[i]==0 的下标 i,返回 i+1 即为结果,此时说明 A 中未出现的最小正整数在 1~n 之间。若 B[i]全部不为 0,返回 i+1(跳出循环时 i=n,i+1 等于 n+1),此时说明 A 中未出现的最小正整数是 n+1。

3) 时间复杂度:遍历 A 一次,遍历 B 一次,两次循环内操作步 为 O(1)量级,因此时间复杂度为 O(n)。空间复杂度:额外分配了 B[n],空间复杂度为 O(n)。

## 42. 解析:

1)为了求解最经济的方案,可以把问题抽象为求无向带权图的最小生成树。可以采用手动 prim 算法或 kruskal 算法作图。注意本题最小生成树有两种构造,如下图所示。



方案的总费用为16。

- 2) 存储 中的图可以采用邻接矩阵(或邻接表)。构造最小生成树采用 Prim 算法(或 kruskal 算法)。
- 3) TTL=5,即 IP 分组的生存时间(最大传递距离)为 5,方案 1 中 TL 和 BJ 的距离过远,TTL=5 不足以让 IP 分组从 H1 传送到 H2,因此 H2 不能收到 IP 分组。而方案 2 中 TL 和 BJ 邻近,H2 可以收到 IP 分组。

#### 43. 解析:

- 1)程序定时向缓存端口查询数据,由于缓存端口大小有限,必 在传输完端口大小的数据时访问端口,以防止部分数据没有被及时读取而丢失。设备 A 准备 32 位数据所用时间为 4B/2MB=2us,所以最多每隔 2us 必 查询一次,每秒的查询次数至少是  $1s/2us=5\times10^5$ ,每秒 CPU 用于设备 A 输入/输出的时间至少为  $5\times10^5\times10\times4=2\times10^7$  个时钟周期,占整个 CPU 时间的百分比至少是  $2\times10^7/500M=4\%$ 。
- 2) 中断响应和中断处理的时间为 400× (1/500M) =0.8us, 这时只需判断设备 B 准备 32 位数据要多久,如果准备数据的时间小于中断响应和中断处理的时间,那么数据就会被刷新、造成丢失。经过计算,设备 B 准备 32 位数据所用时间为 4B/40MB=0.1us,因此,设备 B 不适合采用中断 I/O 方式。
- 3)在 DMA 方式中,只有 处理和后处理需要 CPU 处理,数据的传送过程是由 DMA 控制。设备 B 每秒的 DMA 次数最多为 40MB/1000B=40000,CPU 用于设备 B 输入/输出的时间最多为  $40000\times500=2\times10^7$ 个时钟周期,占 CPU 总时间的百分比最多为  $2\times10^7/500M=4\%$ 。

#### 44. 解析:

- 1) 物理地址由实页号和页内地址拼接,因此其位数为16+12=28;或直接可得20+3+5=28。
- 2) TLB 采用全相联映射,可以把页表内容调入任一块空 TLB 中, TLB 中每 都有一个比

较器,没有映射规则,只要空闲就行。TLB采用静态存储器 SRAM,读写速度快,但成本 ,多用于容量较小的 速缓冲存储器。

3) 图中可以看到, Cache 中每组有两行, 故采用 2 路组相联映射方式。

因为是 2 路组相联并采用 LRU 替换算法, 所以每行(或每组)需要 1 位 LRU 位; 因为采用 回写策略, 所以每行有 1 位修改位(脏位), 根据脏位判断数据是否被更新, 如果脏位为 1 则需要写回内存。

28 位物理地址中 Tag 字段占 20 位,组索引字段占 3 位,块内偏移地址占 5 位,故 Cache 共有  $2^3$ =8 组,每组 2 行,每行有  $2^5$ =32B;故 Cache 总容量为  $8\times2\times(20+1+1+1+32\times8)$  =4464 位=558 字节。

Cache 中有效位用来指出所在 Cache 行中的信息是否有效。

4)虚拟地址分为两部分:虚号、内地址;物理地址分为两部分:实号、内地址。利用虚拟地址的虚号部分去查找TLB表(缺失时从表调入),将实号取出后和虚拟地址的内地址拼接,就形成了物理地址。虚号008CH恰好在TLB表中对应实号0040H(有效位为1,说明存在),虚拟地址的后3位为内地址040H,则对应的物理地址是0040040H。

物理地址为 0040040H, 其中 20 位 00400H 为标志字段, 低 5 位 00000B 为块内偏移量, 中间 3 位 010B 为组号 2, 因此将 00400H 与 Cache 中的第 2 组两行中的标志字段同时比较,可以看出,虽然有一个 Cache 行中的标志字段与 00400H 相等,但对应的有效位为 0,而另一 Cache 行的标志字段与 00400H 不相等,故访问 Cache 不命中。

因为物理地址的低 12 位与虚拟地址低 12 位相同,即为  $0010\,0110\,0000$ B。根据物理地址的结构,物理地址的后八位 01100000B 的前三位 011B 是组号,因此该地址所在的主存映射到 Cache 组号为 3。

### 45. 解析:

1)由图可知,地址总长度为32位, 20位为虚 号,低12位为 内地址。且虚 号 10位为 目录号,低10位为 号。展开成二进制则表示为:



故十六进制表示为 0180 6008H

- 2) PDBR 为 目录基址地址寄存器(Page-Directory Base Register),其存储 目录表物理内存基地址。进程切换时,PDBR 的内容会变化;同一进程的线程切换时,PDBR 的内容不会变化。每个进程的地址空间、 目录和 PDBR 的内容存在一一对应的关系。进程切换时,地址空间发生了变化,对应的 目录及其起始地址也相应变化,因此需要用进程切换后当前进程的 目录起始地址刷新 PDBR。同一进程中的线程共享该进程的地址空间,其线程发生切换时,地址空间不变,线程使用的 目录不变,因此 PDBR 的内容也不变。
- 3) 改进型 CLOCK 置换算法需要用到使用位和修改位,故需要设置访问字段(使用位)和 修改字段(脏位)。

46. 解析:

- 1)簇大小为 4KB,每个地址 长度为 4B,故每簇有 4KB/4B=1024 个地址 。最大文件的 物理块数可达  $8+1\times1024+1\times1024^2+1\times1024^3$ ,每个物理块(簇)大小为 4KB,故最大文件长度为  $(8+1\times1024+1\times1024^2+1\times1024^3)\times4$ KB=32KB+4MB+4GB+4TB
- (2) 文件索引节点总个数为 1M×4KB/64B=64M, 5600B 的文件占 2 个簇, 512M 个簇可存放的文件总个数为 512M/2=256M。可表示的文件总个数受限于文件索引节点总个数, 故能存储 64M

# 个大小为 5600B 的图像文件

(3) 文件 F1 大小为  $6KB < 4KB \times 8 = 32KB$ ,故获取文件 F1 的最后一个簇的簇号只需要访问索引节点的直接地址项。文件 F2 大小为 40KB, $4KB \times 8 < 40KB < 4KB \times 8 + 4KB \times 1024$ ,故获取 F2 的最后一个簇的簇号还需要读一级索引表。综上,需要的时间不相同。

## 47. 解析:

1)广播地址是网络地址中主机号全 1 的地址(主机号全 0 的地址,代表网络本身)。销售部和技术部均分配了 192.168.1.0/24 的 IP 地址空间,IP 地址的前 24 位为子网的网络号。于是在后 8 位中划分部门的子网,选择前 1 位作为部门子网的网络号。令销售部子网的网络号为 0,技术部子网的网络号为 1,则技术部子网的完整地址为 192.168.1.128;令销售部子网的主机号全 1,可以得到该部门的广播地址为 192.168.1.127。

每个主机仅分配一个 IP 地址,计算目前还可以分配的主机数,用技术部可以分配的主机数,减去已分配的主机数,技术部总共可以分配计算机主机数为  $2^7$ -2=126(减去全 0 和全 1 的主机号)。已经分配了 208-129+1=80 个,此外还有 1 个 IP 地址分配给了路由器的端口(192.168.1.254),因此还可以分配 126-80-1=45 台。

2) 判断分片的大小,需要考虑各个网段的 MTU,而且注意分片的数据长度必须是 8B 的整数倍。由题可知,在技术部子网内,MTU=800B,IP 分组头部长 20B,最大 IP 分片封装数据的字节数为 (800-20)/8 × 8=776。至少需要的分片数为 (1500-20)/776 = 2。第 1 个分片的偏移量为 0;第 2 个分片的偏移量为 776/8=97。