

解析第一版，不排除答案、解析（特别是综合题），存在一些错误和问题。

一、单项选择题

1. C

按上三角存储， $m_{7,2}$ 对应的是 $m_{2,7}$ ，在它之前有：

第 1 列：1

第 2 列：2

.....

第 6 列：6

第 7 列：1

前面一共 $1+2+3+4+5+6+1$ 个元素，共 22 个元素，数组下标从 0 开始，故下标为 $m_{2,7}$ 的数组下标为 22。

2. D

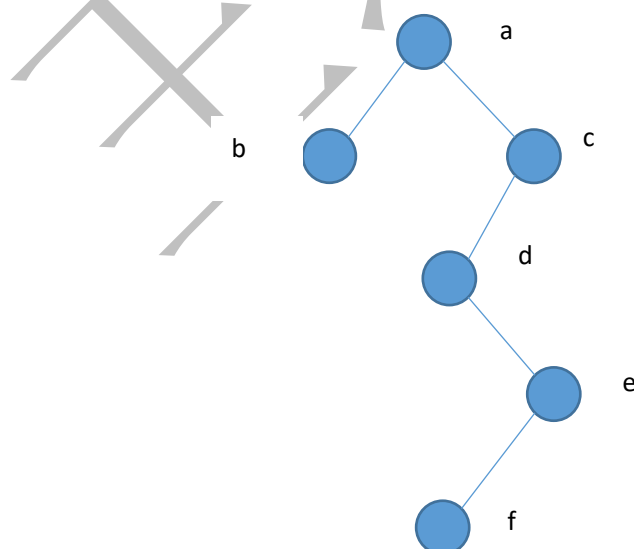
第一个 pop 栈中状态为 a,b，pop 出栈元素为 b，第二个 pop 栈中状态为 a,c，pop 出栈元素为 c，第三个 pop 栈中状态为 a,d,e，pop 出栈元素为 e，把序列连起来就是 b,c,e。

3. A

由于题目明确说明只存储结点数据信息，所以采用顺序存储时要用数组的下标保存结点的父子关系，所以对于这棵二叉树存储的结果就是存储了一棵五层的满二叉树，五层的满二叉树结点个数为 $1+2+4+8+16=31$ ，所以至少需要 31 个存储单元。

4. C

1. 森林的先根遍历对应它自己转化后二叉树的先序遍历，森林的后根遍历对应它自己转化后二叉树的中序遍历，所以先根和后根可以唯一确定森林转化后的二叉树，如下：



后序遍历为：b,f,e,d,c,a

5. B

在 4, 5, 1, 2, 3 中由于 1 先插入, 所以 1 会成为 4 的左孩子, 2 会成为 1 的右孩子……。

6. B

首先有直觉这个方式输出的肯定是一个逆序, 所以用一个简单的例子测试一下 $a \rightarrow b \rightarrow c$, 按照题目的遍历方式得到的序列为 c, b, a , 排除 A, C, D, 选 B

下面严谨的论证一下, 题目已经限定有向无环图, 假设从 a 结点出发开始深度遍历, 那么这一次递归到最大深度, 必然终止于某结点(记为 h 结点), h 结点必然没有出度。此时 h 输出, 程序栈退栈, 回到 h 的前一个结点(记为 f), 如果 f 还有其他出度, 那么此时要访问其他出度, 直到每一个出度的分支都访问结束才能访问 f , 这样来看, 一个结点要被访问的前提必须是他的所有出度分支都要被访问, 换句话说也就是等一个结点没有出度时才可以访问, 这就是逆拓扑排序(每次删除的都是出度为零的结点)。

7. A

先将所有边按权值排序, 然后依次取权值最小的边但不能在图中形成环, 此时取得权值序列为 5, 6, 此时 7 不能取因为形成了环, 接下来去 9, 10, 11, 按权值对应的边。

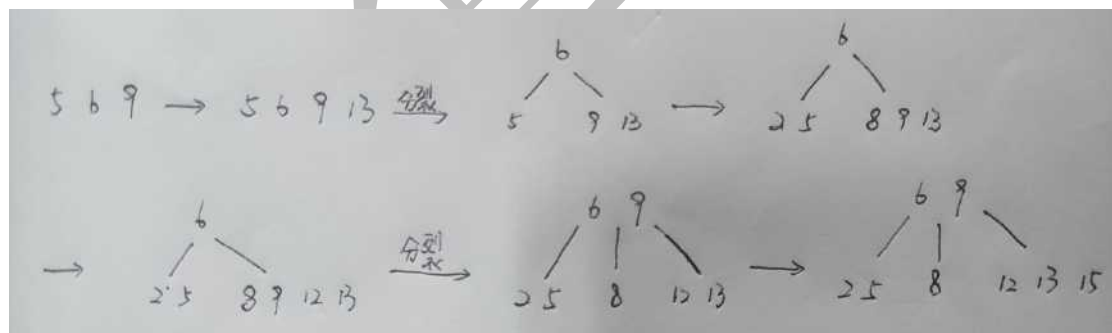
8. B

A 改为权值之和最大的路径, B 的最长就是指权值之和最大, C 增加关键活动一定会增加工期, D 减小一关键活动不一定会缩短工期

9. 回忆不全

III 错误, 因为堆只要求根大于左右子树, 并不要求左右子树有序。

10. B



11. A

直接插入排序在有序数组上的比较次数为 $n-1$, 简单选择排序的比较次数为 $1+2+\dots+n-1=n(n-1)/2$ 。II, 辅助空间都是 $O(1)$, 没差别, III, 因为本身已经有序, 移动次数均为 0。

12. B

王道书 P13 原话 “机器字长通常与 CPU 的寄存器位数、加法器有关”。

13. A

展开 1100 1000 0000 0000 ... (省略 16 个 0) H, 将其转换为对应的 float 或 int。

如果是 float，尾数是隐藏了的最高位 1，数符为 1 表示负数，阶码 $1001\ 0000=2^7+2^4=128+16$ ，减去偏置值 $127=17$ ，为 -2^{17} ；
如果是 int，带符号补码，为负数，数值部分取反加 1， $011\ 1000\ 0000\ 0000\ \dots$ （省略 16 个 0）H，算出值为 $-7*2^{27}$ 。

14. D

根据按边界对齐和小端方式的定义(王道书第 2 章和第 4 章的常见问题分别给出了大小端存放和按边界对其的定义及举例)，给出变量 a 的存放方式如下：

x1 (LSB)	x2 (MSB)	null	null
00H	00H	34H	12H

首地址为 2020 FE00H，按字节编址，则 34H 所在单元地址为 $2020\ FE00H + 6 = 2020\ FE06H$ 。

15. D

Cache 由 SRAM 组成。TLB 通常用相联存储器组成，也可以由 SRAM。DRAM 需要不断的刷新，性能较低，肯定不能选。

16. A

48 条指令需要 6 位操作码字段，4 种寻址方式需要 2 位寻址特征位，故寻址范围为 $0\sim 255$ 。注意，主存地址不能为负数。

17. B

II 肯定错误，IV 是通过增加功能部件实现的并行。
在理想情况下，I 单周期 CPU，令指令周期=时钟周期；III 基本流水线 CPU，让每个时钟周期流出一条指令（执行完一条指令）。

18. A

自陷是属于内中断。

19. C

每个时钟周期传递 2 次，根据公式， $2.4G*2*2*2B/s=19.2GB/s$ ，选 C。
误区，公式里最后已经乘了 2 次了（全双工），大家在求时不要再乘了。

20. 题干不完整

II 外部中断，描述的是时钟中断；III 外部中断，外部事件。

21. B

容易误选 A，这点王道书上有描述 P278。
B. CPU 响应中断需要满足 3 个条件，王道书 P278。

22. C

注意是周期挪用方式，如果是停止 CPU 访存，那就是应该选 A。
本题也有观点认为选 A。

23. B

既可以是读的方式，也可以是写的方式，A 错误。系统打开文件表整个系统只有一张，同一个文件打开多次只需要改变引用计数，不需要对应多项，B 正确。用户进程的打开文件表关于同一个文件不一定相同，C 错误。进程关闭文件时，文件的引用计数减少 1，引用计数变为 0 时才删除，D 错误。

24. A

链接分配不能支持随机访问，B 错误。连续分配不支持可变文件长度，C 错误。动态分区分配是内存管理方式非磁盘空间管理方式，D 错误。

25. D

中断的保存硬件和软件分别都要保存部分寄存器内容，硬件保存程序计数器 PC，操作系统保存程序状态字 PSW，不仅仅由操作系统单独完成，I 错误。

26. D

多级反馈队列调度需要综合考虑优先级数量、优先级之间的转换规则等，I,II,III,IV 均正确。

27. B

此道题作出需求矩阵 $NEED = MAX - ALLOCATED$ 即可。

$$\text{Need} = \text{Max} - \text{Allocated}$$
$$= \begin{bmatrix} 4 & 4 \\ 3 & 1 \\ 3 & 4 \end{bmatrix} - \begin{bmatrix} 2 & 3 \\ 2 & 1 \\ 1 & 2 \end{bmatrix} = \begin{bmatrix} 2 & 1 \\ 1 & 0 \\ 2 & 2 \end{bmatrix}$$

同时，由 allocated 矩阵得知当前 available 为 (1, 0)。由需求矩阵可知，初始只能满足 p2 需求。P2 释放资源后 available 变为 (4, 1)。此时仅能满足 p1 需求，p1 释放后可以满足 p3。

故得到顺序 p2->p1->p3。B 正确。

28. D

I 影响缺页中断发生的频率；II,IV 影响缺页中断的处理时间；II 影响访问慢表和访问目标物理地址的时间，故 I,II, III,IV 均正确。

29. B

父进程可以和子进程共享一部分共享资源，但是不和子进程共享虚拟地址空间，在创建子进程时，会为子进程分配空闲的进程描述符、唯一标识的 pid 等，B 错误。

30. D

设备可以看作特殊文件，A 正确。B 为知识点，正确。访问设备的驱动程序与具体设备无关，D 错误。

31. B

最多创建文件个数=最多索引节点个数。由题，索引节点占 4 个字节，对应 32 位，最多可以表示 2^{32} 个文件，B 正确。

32. C

I,II,III 分别符合互斥、空闲让进、有限等待的原则，不能立即进入临界区的进程可以选择等待部分时间，IV 错误。故 C 正确。（个人认为 IV 也对，四个原则：II 是空闲让进，I 是忙则等待，III 是有限等待，IV 是让权等待。不排除是回忆版真题有误）

33. C

王道书 P14，网络协议主要由语义、语法和时序（一般教材定义为同步）三部分组成，即协议三要素。语义：规定通信双方彼此“讲什么”，规定所要完成的功能，如规定通信双方要发出什么控制信息，执行的动作和返回的应答。语法：规定通信双方彼此“如何讲”，即规定传输数据的格式，如数据和控制信息的格式。时序：或称同步，规定了信息交流的次序。由图可知发送方与接收方依次交换信息，体现了协议三要素中的时序要素。

34. B

虚电路服务需要有建立连接过程，每个分组使用短的虚电路号，属于同一条虚电路的分组按照同一路由进行转发，分组到达终点的顺序与发顺顺序相同，可以保证有序传输，不需要为每条虚电路预分配带宽。

35. C

网络层设备路由器可以隔离广播域和冲突域，链路层设备普通交换机只能隔离冲突域，物理层设备集线器、中继器既不能隔离冲突域也不能隔离广播域。题中共有 2 个广播域，4 个冲突域。

36. D

发送数据帧和确认帧的时间分别为 800ms, 800ms。发送周期为 $T=800+200+800+200=2000\text{ms}$ 。采用停止-等待协议，信道利用率为 $800/2000=40\%$ 。

37. A

为了尽量避免碰撞，802.11 规定，所有的站在完成发送后，必须再等待一段很短的时间(继续监听)才能发送下一帧。这段时间通称为帧间间隔 IFS (InterFrame Space)。帧间间隔的长短取决于该站要发送的帧的类型。IEEE 802.11 推荐使用 3 种帧间隔 (IFS)，以便提供基于优先级的访问控制。DIFS(分布式协调 IFS)：最长的 IFS，优先级最低，用于异步帧竞争访问的时延。PIFS (点协调 IFS)：中等长度的 IFS，优先级居中，在 PCF 操作中使用。SIFS (短 IFS)：最短的 IFS，优先级最高，用于需要立即响应的操作。网络中的控制帧以及对所接收数据的确认帧都采用 SIFS 作为发送之前的等待时延。当结点要发送数据帧时，载波监听到信道空闲时，需等待 DIFS 后发送 RTS 预约信道，IFS1 对应的帧间隔 DIFS，时间最长，图中 IFS2，IFS3，IFS4 对应 SIFS。

38. C

在不出现拥塞的前提下，拥塞窗口从 8KB 增长到 20KB 所需的最长时间（由于慢开始门限可以根据需求设置 所以这里面为了求最长时间 可以假定在慢开始门限小于等于 8KB，这样

由 8KB-20KB 的过程中都是加法增大), 考虑拥塞窗口达到 8KB 时, 以后的每个轮次拥塞窗口逐次加 1, 需 $12 \times 2 = 24\text{ms}$ 后达到 20KB 大小。

也有同学反馈: 题目是从 8k-32k, 那么答案就选 D。

39. C

主机甲与主机乙建立 TCP 连接时发送的 SYN 段中的序号为 1000, 则在数据数据传输阶段所用序号起始为 1001, 在断开连接时, 甲发送给乙的 FIN 段中的序号为 5001, 在无任何重传的情况下, 甲向乙已经发送的应用层数据的字节数为 $5001 - 1001 = 4000$ 。

40. D

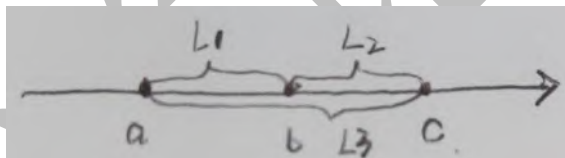
忽略各种时延情况下, 最短时间, 即本地域名服务器存在域名与 IP 地址映射关系, 仅需主机向本地域名服务器递归查询一次 10ms, 传送数据 10ms, 最短时间共需 20ms; 最长时间即本地域名服务器不存在域名与 IP 地址映射关系, 需向本地域名服务器递归查询一次后, 迭代查询各级域名服务器 3 次, 需 40ms, 传送数据 10ms, 最长时间共需 50ms。

二、综合应用题

41 解析:

由 $D = |a-b| + |b-c| + |c-a|$ 得 D 最小为 0, 此时 $a=b=c$, 这就是最小情况, 增大 a, b, c 任意一个都会使 D 变大, 所以我们要找的就是 S1, S2, S3 中值最接近的一组数 (三个数), 同时改变 a, b, c 求最小的 D 是一个复杂组合优化问题, 为了简化问题, 我们每次只改变 a, b, c 中的一个值观察 D 的变化情况。

换个角度看, 分析 $D = |a-b| + |b-c| + |c-a|$, 不失一般性的假设 $a \leq b \leq c$, 观察下面数轴



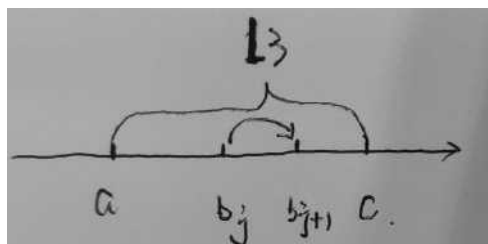
$|a-b| = L1$, $|b-c| = L2$, $|c-a| = L3$, $D = |a-b| + |b-c| + |c-a| = L1 + L2 + L3 = 2L3$, 事实上决定 D 大小的关键在于 a 和 c 的距离, 问题就变为了每次固定 c 找一个 a 使得 $|c-a| = L3$ 最小, 然后更新 c 寻找下一个 a。

(1) 算法设计思想: 不失一般性的假设 $a \leq b \leq c$ 。分情况讨论:

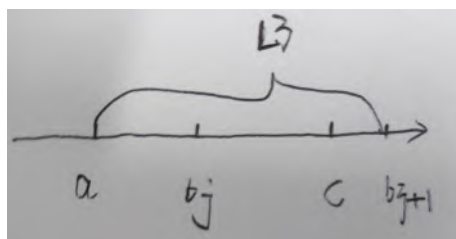
I. $a_i \leq b_j \leq c_k$ 且 $! (a_i = b_j = c_k)$ (a, b, c 不全相同):

此时 $D_{\min} = D_c = |a_i - b_j| + |b_j - c_k| + |c_k - a_i|$, 是一个不为 0 的数, 要使这个数变小, 可能会有两种选择增大 a 或者增大 b (显然增大 c 只会让 D 增加), 即 a_i 变为 a_{i+1} 或 b_j 变为 b_{j+1} 。

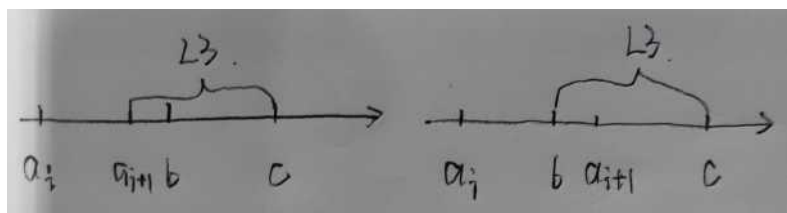
先看 b_{j+1} , ①如果 $a_i < b_{j+1} \leq c_k$, 看数轴, b 其实就是在 a 和 c 之间改变, 并不影响 $L3 = |c_k - a_i|$, 那么此时 D 是不改变的;



②如果 $a_i < c_k < b_{j+1}$, 看数轴, 此时的 $L3 = |a_i - b_{j+1}|$, $L3$ 变大了, 那么此时 D 变大。



再看 a_{i+1} , ①如果 $a_{i+1} < c_k$, 看数轴, 由于 a 增大 $L3 = \max\{|b_j - c_k|, |c_k - a_{i+1}|\} < |c_k - a_i|$, $L3$ 变小了, 此时 D 减小;



②如果 $b_j < c_k < a_{i+1}$, 那么此时 D 变小变大不确定。

综上, 只有增加 a 才可能使 D 变小, 所以 a 变为 a_{i+1} , 重新计算 $D = D_N$, 如果 $D_N < D_{min}$, 则 $D_{min} = D_N$, 循环上面步骤, 直到任一数组遍历结束或出现 II 状态, 最后返回 D_{min} 。

II. $a_i = b_j = c_k$: 这种状态已经是最佳状态, 直接返回 $D_{min}=0$;

(2) 算法代码:

```
int my_abs(int a){
    if(a < 0) return -a;
    else return a;
}

int shortest_dis(int a[], int len_a, int b[], int len_b, int c[], int len_c) {
    int min_d = 0x7fffffff;
    int index_a = 0;
    int index_b = 0;
    int index_c = 0;

    while ((index_a < len_a) && (index_b < len_b) && (index_c < len_c)){
        int temp_a = a[index_a];
        int temp_b = b[index_b];
        int temp_c = c[index_c];
        int d = my_abs(temp_a - temp_b) + my_abs(temp_b - temp_c) + my_abs(temp_c - temp_a);
        if(d < min_d){
            min_d = d;
            if(min_d == 0) break;
        }
        if (temp_a <= temp_b && temp_a <= temp_c) index_a++;
        else if (temp_b <= temp_a && temp_b <= temp_c) index_b++;
        else if (temp_c <= temp_a && temp_c <= temp_b) index_c++;
        else break;
    }

    return min_d;
}
```

(3) 将三个序列的长度分标记为 $L1, L2, L3$, 时间复杂度为 $O(L1 + L2 + L3) = O(\max(L1, L2, L3))$, 空间复杂度为 $O(1)$

42. 解析:

(1) 哈夫曼树。(书上哈夫曼树应用的例子就是前缀编码)

(2) 译码过程: 从前到后依次取 0/1 串的每一位, 如果为 0 就进入左子树, 如果为 1 就进入右子树, 直到整个 0/1 串遍历结束, 遇到叶子节点, 该叶子节点代表的字符串就是该 0/1 串的字符串, 如果没有遇到叶子结点或者遇到了叶子结点但 0/1 串未遍历结束, 则

该 0/1 串所代表的字符串不在该字符集中。

(3) 若某字符集的不等长编码不能确定一颗哈夫曼树，则说明该字符集的不等长编码不具有前缀特性

43.解析

(1) 乘法运算是可以通过加法和移位来实现的。

(2) (3) (4) 组成助教明天考试，后续再提供。

44.解析

(1) $\text{Tag} = 32 - 6 - (9 - 3) = 20$ 位； $\text{LRU} = 3$ 位； 直写法不存在修改位

(2) 当 S 的初始地址不在第三组时， $1024/16=64$ 次；在第三组时，1024 次

(3) 访存的过程和王道书 P121 的描述相似

——组成助教明天考试，后续再提供详细版本。

45.解析

本题要求实现操作的先后顺序，属同步问题。

分别设置信号量 ABCDE 对应 5 个操作，初值都为 0。

5 个进程的动作可描述为：

```
A(){
    完成动作 A;
    V(A);
}
B(){
    完成动作 B;
    V(B);
}
C(){
    //C 必须在 A、B 都完成后才能完成
    P(A);
    P(B);
    完成动作 C;
    V(C);
}
D(){
    完成动作 D;
    V(D);
}
E(){
    //E 必须在完成 C、D 之后执行
    P(C);
    P(D);
    完成动作 E;
    V(E);
}
```


}

46.解析

(1) ①页面大小 $= 2^{12} \text{B} = 4096 \text{B} = 4 \text{KB}$ 。每个数组元素 4B，每个页面可以存放 $4 \text{KB} / 4 \text{B} = 1024$ 个数组元素，刚好是数组的一行。1080 000H 的虚页号为 1080H，因此 a[0] 行存放在虚页号为 1080H 的页面中，a[1] 行存放在页号为 1081H 的页面中。a[1][2] 的虚拟地址为 $1081\ 000\text{H} + 4 \times 2 = 1081\ 008\text{H}$ 。

②转换为二进制 000100 0010000001 000000001000，根据虚拟地址结构可知，对应的页目录号为 4，页号为 129。

③进程的页目录表起始地址为 0020 1000H，每个目录项长 4B，因此 4 号页目录项的物理地址是 $0020\ 1000\text{H} + 4 \times 4 = 0020\ 1010\text{H}$ 。

④页目录项存放的页框号为 00301H，该页框的起始地址为 00301 000H，因此 a[1][2] 所在页的页号为 129，每个页表项 4B，因此对应的页表项物理地址是 $00301\ 000\text{H} + 129 \times 4 = 00301\ 000\text{H} + 204\text{H} = 00301\ 204\text{H}$ 。

(2) 数组 a 在虚拟地址空间中所占区域必须连续，在物理地址空间中所占区域可以不连续。

(3) 按行遍历的局部性更好，因为“按行有限方式存放”意味着同一行内的所有数据元素都存放在同一个页面中（页面大小 4096B 刚好可以存放一整行的数据元素）。遍历同一行的所有元素访问的都是同一个页面，同一列的各个元素都存放在不同的页面中。

47 解析：

(1) 路由器 R2 开启 NAT 服务，当路由器 R2 从 WAN 口收到来自 H2 或 H3 发送过来的数据根据 NAT 转换表发送给 WEB 服务器对应端口。R2 的 NAT 转换表可设置如下：

外网		内网	
IP 地址	端口号	IP 地址	端口号
203.10.2.6	默认端口号	192.168.1.2	80
203.10.2.6	默认端口号	192.168.1.3	80

(2) H2 发送 P 的源 IP 地址：192.168.1.2 目的 IP 地址 203.10.2.2

R3 转发后 P 的源 IP 地址：203.10.2.6 目的 IP 地址：203.10.2.2

R3 转发后 P 的源 IP 地址：203.10.2.6 目的 IP 地址：192.168.1.2