重庆大学2018年硕士研究生入学考试试卷详细解析

解析如有异议，请联系重庆大学 2019 年计算机学院考研 VIP 群 234147197，负责相应科目的学长学姐！

考研是一场修行，越努力越幸运，祝大家考研成功！

-----砍柴学长 2018.12.03

1. 【答案】C

【解析】根据顺序表和链表的特点，顺序表最大的特点是根据下标快速定位元素，因此答案选择 C。

2. 【答案】C

【解析】栈和队列的特点是在端点处操作元素，但栈是后进先出的特点，而队列是先进先出。因此答案选择 C。

3. 【答案】C

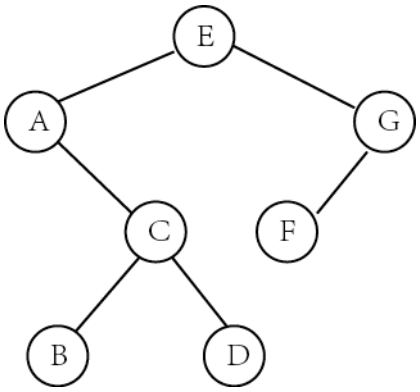
【解析】题目要求在指针 p 和 q 所指节点之间插入指针 a 所指节点，答案 B 符合要求。答案 A，B，D 不符合要求。

4. 【答案】C

【解析】二叉树第 i 层最多可以有 2^(i-1)个节点。答案选 C。

5.【答案】B

【解析】根据中序和后序序列，可以得出二叉树如下，因此前序序列为答案 B。



6.【答案】A

【解析】二叉排序树的中序遍历是一个递增序列，答案选 A。

7.【答案】A

【解析】无向连通图的生成树是该连通的一个极小连通子图。因此答案选 A。

8.【答案】B

【解析】n 个节点的无向图的边数最多为 n*(n-1)/2，n 个节点的有向图的边数最多为无向图的两倍，因此答案选择 B。

9【答案】C

【解析】根据折半查找的特点，对比的元素依次是 15,23,29,27。因此需要比较 4 次。

10.【答案】C

【解析】快速排序最好的情况是序列无序，每次将序列均匀划分为两半，时间复杂度为 O(n*log n)。但是放序列有序的时候，不能将序列均分，排序的时间最长，时间复杂度为 O(n^2)；冒泡排序的时间复杂度为 O(n^2)；堆排序为时间复杂度为 O(n*log n)；希尔排序的时间复杂度跟选择的步长有关，最坏的时间复杂度为 O(n^1.5)。

11.【答案】B

【解析】从序列中找到前 10 个最小的元素最好的方式是采用堆排序，采用大顶堆。时间复杂度为 O(n*log k)（k 在这里为 10）；快速排序和二路归并排序的时间复杂度为 O(n*log n)；插入排序的时间复杂度为 O(n^2)。

12.【答案】A

【解析】由题意可知，根据完全二叉树特点，当节点数为 n 时，节点 i 的左子树情况：当 2i>n 时，此时节点 i 没有左子树；当节点 2i<=n 时，节点 i 的左子树为 2i。因此节点编号为 49 的左子树编号为 98，答案选 A。

13. 【答案】C

【解析】由每次读出数据 16 位 = 2B，连续启动两次读或写操作所需间隔的最小时间(存储周期)为 250 ns = 250×10^-9 s，因此平均数据传输速率为 2B/(250×10^-9s) = 8×10^6 B/s。**注意：**存储周期与存取时间的区别，后者是 CPU 读或写存储器内数据的过程时间。

14. 【答案】B

【解析】浮点数的表示格式为 N = (r^E)×M。r 为浮点数的基数，通常为 2，E 为阶码，M 为尾数。阶码 E 的位数反映了浮点数的表示范围，尾数 M 的位数反映了浮点数的精度。因此阶码位数越多，浮点数可表示的范围越大；尾数越多，浮点数可表示的精度越高。答案选 B。

15. 【答案】A

【解析】**溢出**有两种情况：当数据小于定点数能表示的最小值时，计算机将它们作 0 处理，称为**下溢**；大于定点数能表示的最大值时，计算机将无法表示，称为**上溢**。两种溢出，本质上都是运算的结果超出了机器的表示范围。B 选项最高数值位进位有可能溢出(如单符号定点数)，借位不会发生溢出，C、D 明显错误。答案选 A。

16. 【答案】D

【解析】IEEE 754 单精度浮点数格式为：1 位数符+8 位阶码+23 位尾数数值。 0.75 = 1/2+1/4，化成二进制为 0.11。该数为正数，数符为 0；尾数为.11，左移一位(阶码 0 减 1 得到-1)隐藏整数 1 来提高尾数精度，得到尾数为.1，进行扩展至 23 位为.100 0000

0000 0000 0000 0000；阶码为-1，加上偏置值 127 得 126，二进制为 0111 1110。因此，0.75 对应的二进制形式为 **0** 0111 1110 100 0000 0000 0000 0000 0000，答案选 D。

17. 【答案】C

【解析】LW 指令格式见《计算机组成与设计：硬件/软件接口 第五版》P175，LW 将主存地址为 Reg[25:21]+offset[15:0]的数据写入到寄存器 Reg[20:16]。显然，操作数的地址为基址寄存器内容+偏移量，寻址方式为基址寻址。答案选 C。

18. 【答案】A

【解析】分析 4 条指令，ADD、BEQ 属于**寄存器-寄存器型**指令，所需时间小于**寄存器-存储器型**指令；LW、SW 属于**寄存器-存储器型**指令，而 LW 访问数据存储器后将数据写入到寄存器需要花费额外时间，故 LW 所花时间大于 SW。因此 LW 指令执行的时间最长，为使所有指令在一个周期都能完成，指令周期应设置为 LW 需要的时间，答案选 A。**注意：**如果在计算题中要求 MIPS 指令周期具体值，则仅需要求出 LW 指令关键路径的执行时间即可，无须求出三种类型指令关键路径执行时间取最长，灵活应对。

19. 【答案】B

【解析】DMA 控制器在传送完成时向 CPU 发送一中断信号，CPU 收到中断信号后执行中断处理程序，中断结束后返回被中断程序。中断的目的是通知 CPU 数据块传输结束，需要 CPU 执行中断服务程序进行 DMA 结束处理(例如，校验数据是否正确，决定是否继续传送等)。答案选 B。

20. 【答案】D

【解析】为使流水线每个阶段的操作都能顺利完成，每个流水段的时间应设置为时间最长的流水段的时间，答案选 D。

21. 【答案】B

【解析】指令寄存器 IR、存储器数据寄存器 MDR 和存储器地址寄存器 MAR 是 CPU 的内部工作寄存器，程序员无法直接获取和设置它们的值。汇编语言程序员通过汇编程序可以对某个寄存器进行访问，因此对于通用寄存器，程序员可以使用，答案选 B。

22. 【答案】C

【解析】这里考察微程序的基本概念。**微程序基本思想：**每条机器指令由一个微程序实现，每个微程序包含若干微指令，每条微指令对应若干微命令(or 微操作)。程序放在主存储器中，微程序放在控制存储器中。分析 4 个选项，答案选 C。

23. 【答案】D

【解析】参考《计算机组成与设计：硬件/软件接口 第五版》P175。MIPS 32 系统中，BEQ 指令偏移量字段位数为 16 位，可表示的范围为 64K，并且偏移量是以指令字为单位(见黑皮书 P169)，即 4 个字节。因此，偏移量单位换算成字节，地址范围为分支前后 128K，答案选 D。

24. 【答案】D

【解析】采用独立的指令存储器和数据存储器，主要目的是为了避免发生结构相关(资源冲突)，提高指令流水线执行的并行度。结构相关发生的 2 个条件：同一时刻争用同一资源。采用独立的指令存储器和数据存储器相当于以空间换时间。分析四个选项，A、B 明显不对；C 选项在单个指令周期内可以访问存储器两次(比如，取指周期取指令，执行周期取操作数)；D 选项采用独立的指令/数据存储器速度更快，答案选 D。

25.【答案】D

【解析】D。操作系统管理计算机系统中的软硬件资源，包括数据资源。如果 C 不肯定。其实用排除法也很好做，操作系统肯定需要管理硬件资源和软件资源，那么又是单选题，必然选 D。

26.【答案】A

【解析】长程调度选择程序或作业进入内存；短程调度选择进程获得 CPU。详见操作系统概念 3.2.2。

27.【答案】C

【解析】轮转法必然是可以抢占的。优先级和最短优先可以设计成抢占或者非抢占的。只有先来先服务肯定是非抢占的。参考操作系统概念 5.3，几种算法讲的很详细。

28.【答案】C

【解析】三种线程模型：一对一，多对一，多对多。只有九个字，即使不深入理解，也很容易背下来。但一定注意分清用户线程和内核线程那个是“一”那个是“多”。(多说一句：用户线程可以看作是一个任务单元的抽象，内核线程则负责实际任务的执行，一个任务可以一个内核线程执行，这就是一对一；一个内核线程可能执行多个任务，这就是多对一；最后多个内核线程执行多个任务，这就是多对多；但是不存在一个任务单元交给多个内核线程执行的情况，所以不存在一对多)。

29.【答案】B

【解析】三个条件：互斥，前进，有限等待。参考操作系统概念 6.2。

30.【答案】A

【解析】操作系统概念 8.1.3 原文：运行时从虚拟地址到物理地址的映射是由被称为内存管理单元(MMU)的硬件设备来完成的。

31.【答案】D

【解析】没有特别可以解释的地方，层次顺序就是如此。不理解可以记住。就本题而言，一个简单的技巧是：设备肯定是硬件最近，逻辑文件系统最远，所以只要确定这两项，就可以直接定位答案为 D。

32.【答案】B

【解析】电梯调度算法，从磁盘的一端移动到另一端，扫过每个柱面并响应请求，到达一端后，调转方向。（假设给出端点，注意考虑端点，此处未给出，故不考虑）。

33. 【答案】B

【解析】POP(Post Office Protocol, POP)协议属于邮件读取协议，现在使用它的第 3 个版本，即 POP3。当用户读邮件时，用户代理向邮件服务器发出请求，“拉”取用户邮箱中的邮件；MIME(Multipurpose Internet Mail Extensions, MIME)协议在 SMTP 基础上增加了邮件主体结构并定义了传送非 ASCII 码(如中文、俄文等)的编码规则；

IMAP(Internet Message Access Protocol， IMAP) 是一个应用层协议，用来从本地邮件客户端（如 Microsoft Outlook、Foxmail）访问远程服务器上的邮件； HTTP(Hyper Text Transport Protocol， HTTP)协议定义了浏览器怎样向万维网 server 请求万维网文档，以及 server 如何将文档传送给浏览器。分析四个选项，答案选 B。

34. 【答案】C

【解析】物理层接口具有的四种特性分别是机械特性、电气特性、功能特性和过程特性。

机械特性：主要定义物理连接的边界点，即接插装置。指明接口所用的接线器的形状和尺寸、引线数目和排列、固定和锁定装置等等。

电气特性：规定物理连接上，线路上信号的电压高低、阻抗匹配、传输速率和距离限制等。

功能特性：指明物理接口各条信号线的用途，比如某条线上出现的某一电平的电压表示何意。

过程特性：主要定义各条物理线路的工作规程和时序关系。

依据定义，本题某个电压范围表示含义，属于功能特性。答案选 C。

35. 【答案】A

【解析】TDM 技术（Time Division Multiplexing，时分复用）在数字通信系统中逐渐得到广泛的应用后，目前，在数字通信系统中存在两种时分复用系统，一种是 ITU-T 推荐的 E1 系统，一种是由 ANSI 的 T1 系统。北美使用的是 T1 系统，我国采用的是欧洲的 E1 标准。

36. 【答案】A

【解析】ping 是一种计算机网络工具，用来测试数据包能否透过 IP 协议到达特定主机。ping 的运作原理是向目标主机传出一个 ICMP（Internet Control Messages Protocol）echo@要求数据包，并等待接收 echo 回应数据包。程序会按时间和成功响应的次数估算丢失数据包率（丢包率）和数据包往返时间（网络时延，Round-trip delay time）。

37. 【答案】D

【解析】OSPF(Open Shortest Path First， OSPF)是内部网关协议 IGP 的一种，使用分布式链路状态路由算法的典型代表，属于网络层协议；TCP(Transmission Control Protocol, TCP)是一种面向连接的、可靠的、基于字节流的传输层通信协议；BGP(Border Gateway Protocol，BGP)是不同自治系统的路由器之间交换路由信息的协议，属于基于 TCP 的应用层协议；RIP(Routing Information Protocol, RIP)也是内部网关协议 IGP，是一种分布式的基于距离向量的路由选择协议，最大的优点就是简单，属于应用层协议。

38. 【答案】C

【解析】IP 组播环中，数据包的目的地址不是一个，而是一组，形成组地址。所有的信息接收者都加入到一个组内，并且一旦加入之后，流向组地址的数据立即开始向接收者传输，组中的所有成员都能接收到数据包。组播组中的成员是动态的，主机可以在任何时刻加入和离开组播组。组播报文的目的地址使用 D 类 IP 地址，地址范围是 224.0.0.0--239.255.255.255。

39. 【答案】D

【解析】UDP(User Datagram Protocol, UDP)协议在网络中它与 TCP 协议一样用于处理数据包，是一种无连接的协议。UDP 为网络层以上和应用层以下提供了一个简单的接口，各有 16bit 的源端口和目的端口用来标记发送和接受的应用进程，因为 UDP 不需要应答，所以源端口是可选的，如果源端口不用，那么置为零。UDP 只提供数据的不可靠传递，采用尽力交付策略，它一旦把应用程序发给网络层的数据发送出去，就不保留数据备份（所以 UDP 被认为是不可靠的数据报协议），UDP 在 IP 数据报的头部仅仅加入了复用和数据校验（字段）。

40. 【答案】C

【解析】NAT（Network Address Translation），是一种在 IP 数据包通过路由器或防火墙时重写来源 IP 地址或目的 IP 地址的技术。通过将专用网络地址转换为公用地址，整个专用网仅需要一个全球 IP 地址即可，解决了 IPv4 的地址空间不足的问题；CIDR（Classless Inter-Domain Routing）是一个用于给用户分配 IP 地址以及在互联网上有效地路由 IP 数据包的对 IP 地址进行归类的方法，可以大幅提供 IP 地址空间的利用率；IGMP（Internet Group Management Protocol)是用于管理网路协议多播组成员的一种通信协议，IP 主机和相邻的路由器利用 IGMP 来创建多播组的组成员；IPv6 Internet Protocol version 6）是网际协议（IP）的最新版本，用作互联网的网络层协议，用它来取代 IPv4 主要是为了解决 IPv4 地址枯竭问题，不过它也在其他很多方面对 IPv4 有所改进。

41. 【解析】

如下图所示构造的哈夫曼树，定义左侧为 0，右侧为 1，由此可知各个字符的哈夫曼编码为：

字符权值为 6 的哈夫曼编码：01110

字符权值为 12 的哈夫曼编码：01111

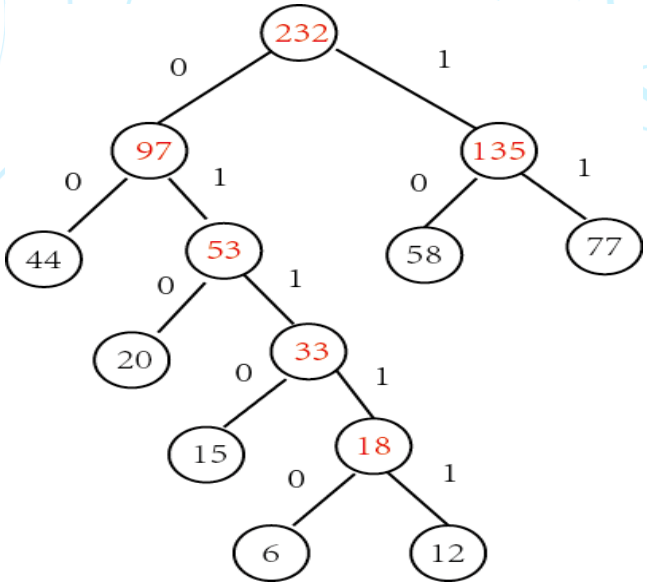
字符权值为 15 的哈夫曼编码：0110

字符权值为 20 的哈夫曼编码：010

字符权值为 44 的哈夫曼编码：00

字符权值为 58 的哈夫曼编码：10

字符权值为 77 的哈夫曼编码：11

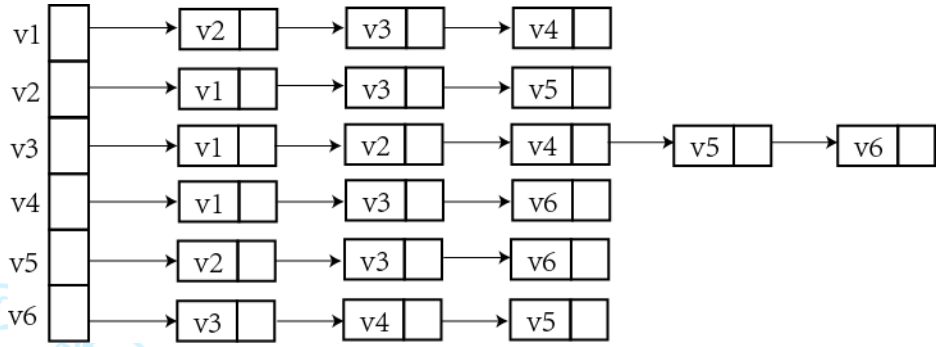


42. 【解析】

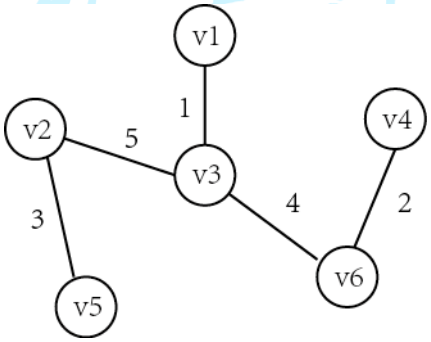
(1)邻接矩阵表示如下图，设邻接矩阵为 A，则：

$$A=\begin{bmatrix}0&6&1&5&0&0\\6&0&5&0&3&0\\1&5&0&5&6&4\\5&0&5&0&0&2\\0&3&6&0&0&6\\0&0&4&2&6&0\end{bmatrix}$$

(2)邻接表表示如下：



(3)使用 prim 算法，将节点 v3 选作起始节点，根据 prim 算法的特点得出的最小生成树如下：



注意:可以选择其它节点作为起始节点，生成树会有所不同。

43. 【解析】

森林转换为二叉树的方法如下：

(1)把每棵树转换为二叉树

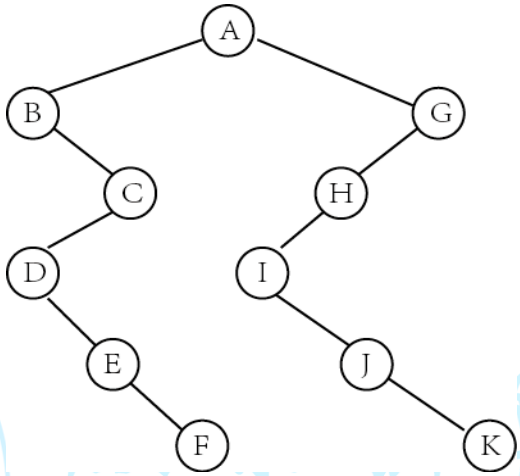
(2)第一棵二叉树不动，从第二棵二叉树开始，依此把后一棵二叉树的根节点作为前一棵的根节点的右孩子。

树转换为二叉树的方法：

①加线。在所有兄弟结点之间加一条线

②去线。树中的每个结点，只保留它与第一个孩子结点的连线，删除其它孩子结点之间的连线

③调整。以树的根节点为轴心，将整个树调整一下(第一个孩子是结点的左孩子，兄弟转过来的孩子是结点的右孩子)



44.【解析】

LRU 根据当前 cache 块向已访问的 cache 序列方向看，选择最久未被访问的 cache 块替换。由题知，cache 总块数为 4。

直接映射：cache 块号 = 主存块号 mod cache 总块数，0 mod 4 = 0，8 mod 4 = 0，6 mod 4 = 2。

访问的内存块	Cache 使用块号	命中情况	访问后 cache 中存放的内容			
			0	1	2	3
0	0	缺失	内存块 0	NULL	NULL	NULL
8	0	缺失	内存块 8	NULL	NULL	NULL
0	0	缺失	内存块 0	NULL	NULL	NULL
6	2	缺失	内存块 0	NULL	内存块 6	NULL
8	0	缺失	内存块 8	NULL	内存块 6	NULL
6	2	**命中**	内存块 8	NULL	内存块 6	NULL

命中率 = 1/6

两路组相联：cache 组号 = 主存块号 mod cache 组数(本题为 2 组)，组间采用直接映射，组内采用全相联映射，cache 组号 0 mod 2 = 0，8 mod 2 = 0，6 mod 2 = 0。Cache 块号 0~1 对应第 1 组，2~3 对应第 2 组。

访问的内存块	Cache 使用块号	命中情况	访问后 cache 中存放的内容			
			0	1	2	3
0	0	缺失	内存块 0	NULL	NULL	NULL
8	1	缺失	内存块 0	内存块 8	NULL	NULL
0	0	**命中**	内存块 0	内存块 8	NULL	NULL
6	1	缺失	内存块 0	内存块 6	NULL	NULL
8	0	缺失	内存块 8	内存块 6	NULL	NULL
6	1	**命中**	内存块 8	内存块 6	NULL	NULL

命中率 = 1/3

全相联映射：主存块映射到 cache 中任何一个空位置。

访问的内存块	Cache 使用块号	命中情况	访问后 cache 中存放的内容			
			0	1	2	3
0	0	缺失	内存块 0	NULL	NULL	NULL
8	1	缺失	内存块 0	内存块 8	NULL	NULL
0	0	**命中**	内存块 0	内存块 8	NULL	NULL
6	2	缺失	内存块 0	内存块 8	内存块 6	NULL
8	1	**命中**	内存块 0	内存块 8	内存块 6	NULL
6	2	**命中**	内存块 0	内存块 8	内存块 6	NULL

命中率 = 1/2

45.【解析】

在主存/cache 系统中，平均存储器访问时间的计算方法有两种：

方法一：《计算机组成与设计：硬件/软件接口 第五版》P272，**同时参考本书假设 cache 和主存异步访问，即若 cache 缺失，再访问主存**。

平均存储器访问时间 = 命中时间 + 缺失率 × 缺失代价；

方法二：唐朔飞版教材算出每种情况下的平均访问时间再相加，类似于求期望。

由题知，时钟周期为 1/2.5GHz = 0.4ns，L1 cache 命中时 CPI 为 1.5，缺失概率为 2%；

L2 cache 命中时 CPI 为 10ns/0.4ns = 25，缺失概率为 0.2%；

主存的访问 CPI 为 150ns/0.4ns = 375，访问概率为 0.2%。

注意：L1 cache 内容是 L2 cache 内容的子集，L2 cache 内容是主存的子集。

（1） 未采用 L2 cache 时，

仅访问 L1 cache 的概率为 98%，同时访问 L1 cache 和主存的概率为 2%

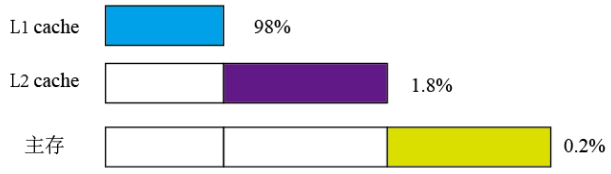
方法一：

平均 CPI = 1.5 + 375 × 2% = 9

方法二：

平均 CPI = 1.5 × 98% + (1.5 + 375) × 2% = 9

（2） 采用 L2 cache 时，仅访问 L1 cache 的概率为 98%，仅同时访问 L1 cache、L2 cache 的概率为 1.8%，同时访问 L1 cache、L2 cache 和主存的概率为 0.2%，



方法一：

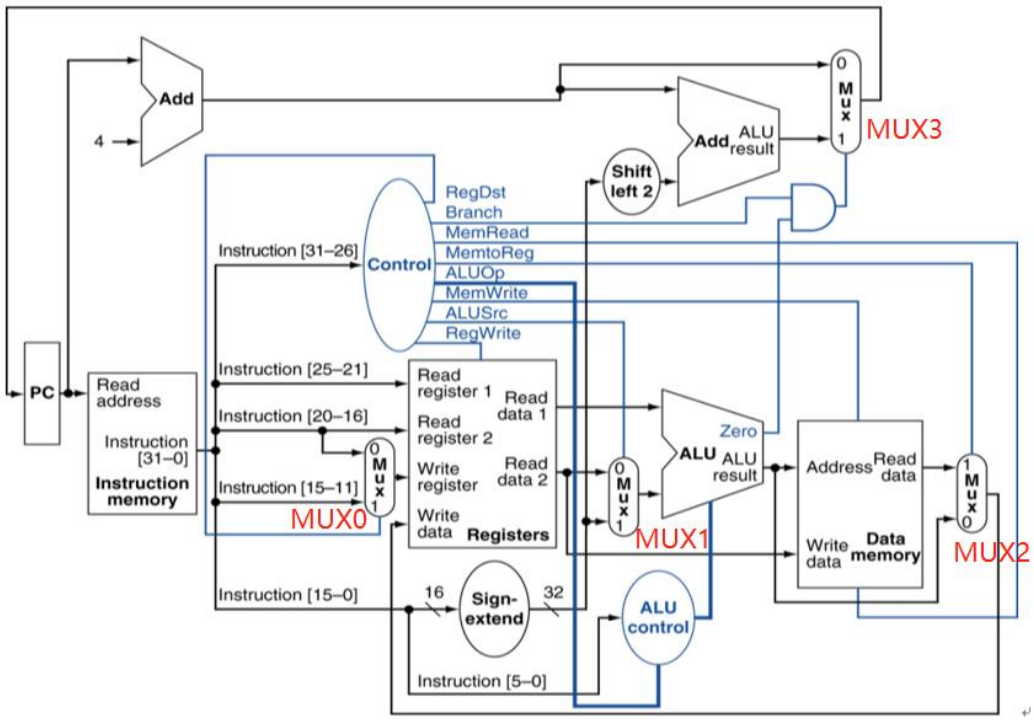
平均 CPI = 1.5 + 25 × 1.8% + (25 + 375) × 0.2% = 2.75

方法二：

平均 CPI = 1.5 × 98% + (1.5 + 25) × 1.8% + (1.5 + 25 + 375) × 0.2% = 2.75

性能提升了 9 / 2.75 - 1 ≈ 2.27。

46.【解析】可参考《计算机组成与设计：硬件/软件接口 第五版》P178。



指令	Branch	ALUSrc	MemtoReg	RegWrite	MemWrite	MemRead
Add	0	0	0	1	0	0
Beq	1	0	0	0	0	0
Lw	0	1	1	1	0	1

分析：

Add 指令：寄存器 Instruction [25-21]和寄存器 Instruction [20-16]的内容相加之和写回寄存器 Instruction [15-11]。

- 非分支指令，branch = 0；
- ALUSrc 控制多选器 MUX1，ALU 第二个源操作数为寄存器 Instruction [20-16]，ALUSrc = 0；
- MemtoReg 控制多选器 MUX2，选择写入寄存器的数据来自 ALU，MemtoReg=0；
- RegWrite 控制寄存器写，Add 结果要写回寄存器 Instruction [15-11]，RegWrite=1；
- MemWrite 和 MemRead 控制 DM 读写，根据定义 Add 指令属于寄存器-寄存器型指令，不访问 DM，MemWrite = 0， MemRead = 0。

Beq 指令：寄存器 Instruction [25-21]和寄存器 Instruction [20-16]的内容相减为 0 则分支。

- 分支指令，branch = 1；
- ALU 第二个源操作数为寄存器 Instruction [20-16]，ALUSrc = 0；
- 不需要写入寄存器操作，MemtoReg=0；
- 不需要写入寄存器操作，RegWrite=0；
- 据定义 Beq 指令属于寄存器-寄存器型指令，不访问 DM，MemWrite = 0， MemRead = 0。

Lw 指令：根据主存地址=寄存器 Instruction [25-21]内容+偏移量 Instruction[15-0]的和从 DM 取数。

- 非分支指令，branch = 0；
- ALU 第二个源操作数为偏移量 Instruction[15-0]，ALUSrc = 1；
- 选择写入寄存器的数据来自 ALU，MemtoReg=1；
- 访存结果要写回寄存器，RegWrite=1；
- 没有存储器写，MemWrite = 0；
- 取数指令要读取 DM 数据，MemRead = 1。

47. 【解析】

(1)在多道程序环境下，多个进程可能竞争一定数量的资源，某个进程申请资源，如果此时资源不可用，那么进程就会进入等待状态。如果所申请的资源被其他进程占用，那么等待进程可能永远无法改变其状态，这种情况称之为死锁。(简略：一组进程相互保持自己资源的同时，等待对方持有的资源，导致进程永远无法推进就是死锁。)

(2)(银行家算法)P1 已经得到全部分配，回收资源；剩余资源(1,1,1)可以满足 P3 请求，P3 执行完之后可以回收资源；剩余资源(2,2,2)可以满足 P4 请求，P4 执行之后回收资源；剩余资源(4,3,2)可以满足 P5 请求；P5 执行完之后回收资源，剩余资源(6,3,4)可以满足 P2 请求。所以，系统安全，安全分配顺序：P1->P3->P4->P5->P2。

(3)有死锁。死锁进程集合为：P2，P3，P5。两种方案：(1)终止所有进程或者一次终止一个进程直到资源满足剩余进程需求；(2)允许资源抢占，逐步从进程中抢占资源给其他进程使用，知道打破死锁。

48. 【解析】

(1)FCFS：(4-0+8-3+15-4+18-5+20-8)/5=9；SJF：(4-0+8-3+20-4+13-5+10-8)/5=7；RR：(4-0+12-3+20-4+17-5+14-8)/5=9.4。

(2)SJF 可能导致饥饿。举例：P3 相比 P4 和 P5 先到达，但是根据 SJF 算法，P3 最后才能

执行，因为等等待队列中 P4 和 P5 任务更短。假如之后源源不断的有新的任务加入等待队列，并且都比 P3 短，那么 P3 将一直等待下去。

49. 【解析】

(1)

OPT：7

页框	1	2	4	3	1	2	5	1	3	2	4	5
帧 0	1	1	1	1	1	1	1	1	1	2	2	2
帧 1		2	2	2	2	2	5	5	5	5	5	5
帧 2			4	3	3	3	3	3	3	3	4	4
缺页	1	1	1	1	0	0	1	0	0	1	1	0

LRU：11

页框	1	2	4	3	1	2	5	1	3	2	4	5
帧 0	1	1	1	3	3	3	5	5	5	2	2	2
帧 1		2	2	2	1	1	1	1	1	1	4	4
帧 2			4	4	4	2	2	2	3	3	3	5
缺页	1	1	1	1	1	1	1	0	1	1	1	1

(2)

OPT：6

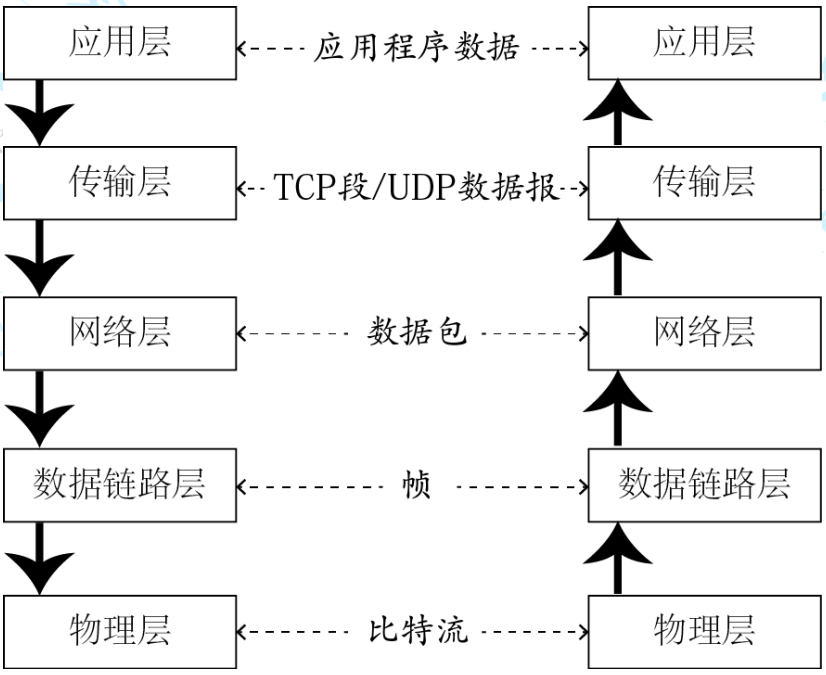
页框	1	2	4	3	1	2	5	1	3	2	4	5
帧 0	1	1	1	1	1	1	1	1	1	1	4	4
帧 1		2	2	2	2	2	2	2	2	2	2	2
帧 2			4	4	4	4	5	5	5	5	5	5
帧 3				3	3	3	3	3	3	3	3	3
缺页	1	1	1	1	0	0	1	0	0	0	1	0

LRU：7

页框	1	2	4	3	1	2	5	1	3	2	4	5
帧 0	1	1	1	1	1	1	1	1	1	1	1	5
帧 1		2	2	2	2	2	2	2	2	2	2	2
帧 2			4	4	4	4	5	5	5	5	4	4
帧 3				3	3	3	3	3	3	3	3	3
缺页	1	1	1	1	0	0	1	0	0	0	1	1

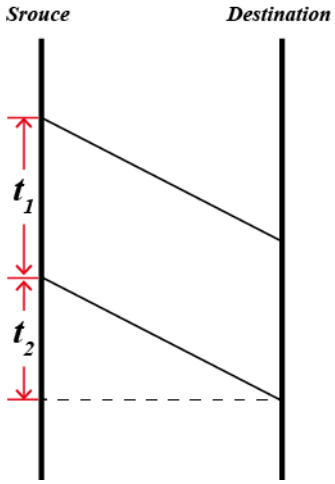
(3)适当的多分配一些页框可以有效减少页面置换次数，尤其是在 OPT 算法无法实现的情况下。

50.【解析】



51.【解析】

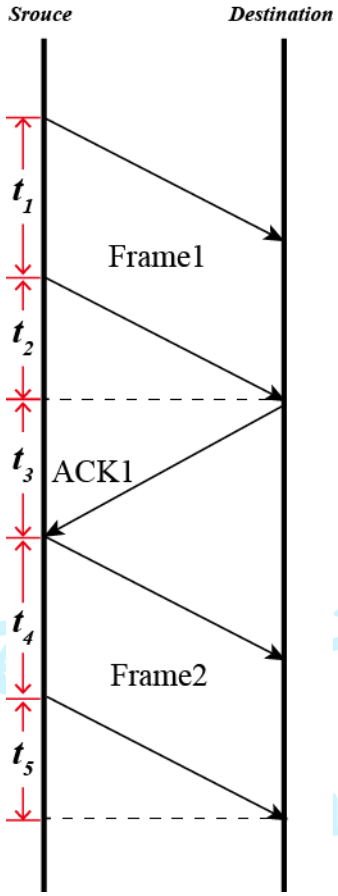
（1）



总时延 = 发送时延t_1 +传输时延 t_2 = 5000bit/1Mbps + 100km/(2*10^8m/s) = 5.5 毫秒

（2）

假设传输过程中没有发生差错，不考虑第二个帧的 ACK 时延。



总的延迟 = 帧 1 的发送时延t_1 + 帧 1 的传输时延 t_2 + 帧 1 的 ACK 时延 t_3 + 帧 2 的发送时延t_4 + 帧 2 的传输时延 t_5 = 2500bit/1Mbps + 100km/(2*10^8m/s) + 100km/(2*10^8m/s) + 2500bit/1Mbps + 100km/(2*10^8m/s) = 6.5 毫秒