

QEOCaSTP

南开软考辅导丛书

全国计算机技术与软件技术资格(水平)考试
指定教程同步练习

网络工程师

命题研究组 编

QUANGUO JISUANJI JISHU YU
RUANJIAN JISHU
ZIGE (SHUIPING) KAOSHI
ZHIDING JIAOCHENG
TONGBU LIANXI
WANGLUO GONGCHENGSHI

南开大学出版社

全国计算机技术与软件技术资格（水平）考试

指定教程同步练习

网络工程师

命题研究组 编

南开大学出版社

天津

内 容 提 要

本书是全国计算机水平考试网络工程师的考前辅导,主要内容有:① 考试要点,概括软件水平考试中本科目涉及的考点以及重点、难点;② 大量典型题以及细致精到的讲解;③ 全真模拟考卷及答案。

本书适用于备战全国计算机水平考试网络工程师的考生以及各类考点培训班。

图书在版编目(CIP)数据

网络工程师 / 命题研究组编. —天津:南开大学出版社, 2009. 8

(全国计算机技术与软件技术资格(水平)考试指定教程同步练习)

ISBN 978-7-310-03215-0

I. 网… II. 命… III. 计算机网络 - 工程技术人员 - 资格考核 - 习题 IV. TP393-44

中国版本图书馆 CIP 数据核字(2009)第 126412 号

版权所有 侵权必究

南开大学出版社出版发行

出版人:肖占鹏

地址:天津市南开区卫津路 94 号 邮政编码:300071

营销部电话:(022)23508339 23500755

营销部传真:(022)23508542 邮购部电话:(022)23502200

*

天津市蓟县宏图印务有限公司印刷

全国各地新华书店经销

*

2009 年 8 月第 1 版 2009 年 8 月第 1 次印刷

787 × 1092 毫米 16 开本 24.375 印张 555 千字

定价:39.00 元

如遇图书印装质量问题,请与本社营销部联系调换,电话:(022)23507125

编委会

主 编：李 波

副主编：许 伟

编 委：杨金魁 刘 欣 张 勇 于樊鹏 李志云 李晓春 王 雷

韦 笑 冯 哲 唐 玮 魏 宇 李 强 张文波 谢 晖

刘 朋 王嘉佳 高 强 邓 卫 李建锋 周 刚

前言

中国计算机技术与软件专业技术资格（水平）考试由国家人事部和信息产业部主办，考试内容涵盖计算机技术与软件的各个主要领域，该考试适应信息技术的迅速发展和更新，鼓励和促进从业人员不断适应和跟进技术变化，满足社会对各种计算机信息技术人才的需要。

本书主要特点

（1）内容针对性强

本书针对水平考试的考点，进行大量典型题的精解，在分析的过程中进行强化训练。我们认为，在考试辅导书中，面面俱到并非一个优势，针对性强才会真正对考生有益。

（2）独具特色的知识点建构方式

每个知识点的复习，是这样建构的：用“考试要点”搭建系统框架，“典型题解”重现重点难点，完成从理论到应用的转变。“全真模拟试卷”从整体上把握考试题型和解答，使考生做到心中有数。

本书主要内容

本书根据水平考试大纲、指定教程以及对历届真题的分析而编写，主要内容有：

- ① 针对每章内容概括考试要点。
- ② “典型题解”讲解细致透彻，考生可以举一反三，相同类型的题目完全可以迎刃而解，通过题目分析和练习，不断加深印象，巩固知识点。
- ③ 模拟考卷给出大量全真模拟题及参考答案，以备战考试。

与我们联系

为了保证本书及时面市和内容准确，很多朋友做出了贡献，李波、许伟、杨金魁、刘欣、张勇、于樊鹏、李志云、李晓春、王雷、韦笑、冯哲、唐玮、魏宇、李强、张文波、谢晖、刘朋、王嘉佳、高强、邓卫、李建锋、周、刚等老师付出了很多辛苦，在此一并表示感谢！

在学习的过程中，您如有问题或建议，请使用电子邮件与我们联系：
book_service@126.com。

全国计算机技术与软件资格（水平）考试命题研究组

目 录

第1章 计算机系统知识	1	第8章 网络软件系统	184
1.1 考试要点	1	8.1 考试要点	184
1.2 典型题解	9	8.2 典型题解	190
第2章 系统开发与运行	46	第9章 网络安全	231
2.1 考试要点	46	9.1 考试要点	231
2.2 典型题解	52	9.2 典型题解	235
第3章 标准化、信息化与知识产权	69	第10章 网络系统设计	253
3.1 考试要点	69	10.1 考试要点	253
3.2 典型题解	73	10.2 典型题解	257
第4章 数据通信基础	86	第11章 计算机专业英语	325
4.1 考试要点	86	11.1 考试要点	325
4.2 典型题解	90	11.2 典型题解	325
第5章 网络体系结构	109	全真模拟试卷(上午试题一)	339
5.1 考试要点	109	参考答案	346
5.2 典型题解	113	全真模拟试卷(下午试题一)	347
第6章 网络分类	130	参考答案	357
6.1 考试要点	130	全真模拟试卷(上午试题二)	361
6.2 典型题解	137	参考答案	368
第7章 网络互连技术	157	全真模拟试卷(下午试题二)	369
7.1 考试要点	157	参考答案	378
7.2 典型题解	163		

第 1 章

考试内容

- 计算机结构。
- 存储器。
- I/O 系统。
- 操作系统知识。

计算机系统知识

1.1 考试要点

计算机系统结构

计算机系统由软件系统和硬件系统两部分组成。软件系统包括系统软件和应用软件，系统软件如操作系统、程序设计语言、编译系统、通用数据库管理系统等；应用软件如文字处理软件、表处理软件等。硬件系统包括主机、存储器、输入/输出设备等，主机主要由中央处理器（CPU）和主存（内存储器）等组成。计算机硬件系统从功能上可分为运算器、控制器、存储器、输入设备和输出设备 5 个部分。

指令系统

指令系统是计算机所有指令的集合。

(1) 复杂指令系统（CISC）：随着硬件成本的不断下降，软件成本的不断提高，使得人们热衷于在指令系统中增加更多、更复杂的指令，来提高操作系统的效率，并尽量缩短指令系统与高级语言的语义差别，以便高级语言的编译和降低软件成本，另外为了保证程序的兼容性，新计算机的指令系统只能增加而不能减少，所以就使得指令系统越来越复杂。

(2) 精简指令系统（RISC）：通过简化指令使计算机的结构更加简单合理，从而提高运算速度。

RISC 计算机的主要特点：

- ① 指令的数量较少。
- ② 复杂的寻址方式很少。
- ③ 指令的长度固定，指令格式种类少。
- ④ 只提供了 Load/Store 指令访问存储器。
- ⑤ 以硬布线逻辑控制为主。
- ⑥ 单周期指令执行。
- ⑦ 拥有相当多的寄存器。
- ⑧ 优化的编译器。

I/O 系统

(1) CPU 控制输入/输出设备工作的方式

CPU 控制输入/输出设备工作的方式有以下 3 种。

① 程序查询输入/输出方式：是指在程序控制下，CPU 与外设之间交换数据。

② 中断输入/输出方式：是指主机在执行程序过程中，遇到了输入/输出中断请求，而中断程序的正常执行，转去对输入/输出中断请求做出处理，待处理完成后，再返回原程序继续执行。

③ 直接存储器 (DMA) 方式：是指高速外设 (一般指磁盘存储器) 与内存之间直接进行数据交换，不通过 CPU 并且 CPU 不对数据交换进行控制。

(2) 系统总线

系统总线是用于连接计算机中各部件 (CPU、内存和外设接口等) 的一组公共信号线。系统总线由数据总线、地址总线和控制总线这 3 组信号线组成。

数据总线用于传输数据，地址总线用于传输地址，控制总线用于传输控制信号。

目前微型机中常用的总线有 4 种，它们是 ISA 系统总线、VESA 局部总线、PCI 局部总线和 MCA 总线。

存储器系统

存储体系结构包括不同层次上的存储器。它们通过适当的硬件、软件有机地组合在一起，形成计算机的存储体系结构。现在一般都将高性能计算机的存储体系结构分为三级存储器结构。三级存储器结构包括高速缓存 (Cache)、主存储器、辅助存储器 (外存储器)。

(1) Cache 存储器

通常在 CPU 和主存储器之间设置有较小容量的高速 Cache 存储器。Cache 容量小但速度快，主存储器速度较低但容量大。通过优化调度算法，系统的性能会大大改善，其存储系统容量与主存相当，而访问速度近似 Cache。在计算机的存储系统体系中，Cache 是访问速度最快的层次。

使用 Cache 改善系统性能的依据是程序的局部性原理，即程序的地址访问流有很强的时序相关性，未来的访问模式与最近已发生的访问模式相似。依据局部性原理，把主存储器中访问概率高的内容存放在 Cache 中，当 CPU 需要读取数据时，就首先在 Cache 中查找是否有所需内容。如果有，则直接从 Cache 中读取；若没有，再从主存中读取该数据，然后同时送往 CPU 和 Cache。

Cache 存储器的组织结构与主存储器不一样，它以行作为基本单元，每一行又分为标志和数据域两部分，数据域中存放着若干项数据，而标志项是这一块数据的地址标识。

当 CPU 发出对存储器的读命令后，其访问地址先送给 Cache 控制器，Cache 检查其地址标识符目录以确定是否匹配项。若发现匹配项 (命中)，则根据其访问地址，确定是对该行数据块中的第几项进行读取，

然后该项即进入 Cache 的数据寄存器。如果没有命中，则到主存储器读取数据。

当 CPU 发出访存请求后，存储器地址先送到 Cache 控制器，以确定所需数据是否已在 Cache 中，若命中，则直接对 Cache 进行访问，这个过程称为 Cache 的地址映射 (Mapping)。常见映射方法有直接映射、相联映射和组成映射。

当 Cache 存储器产生了一次访问失效之后，相应的数据应同时读入 CPU 和 Cache。但当 Cache 已存满数据后，需要通过某种算法淘汰部分数据，从而更新 Cache 和主存的映射关系。常见的映射方法有直接映射、相联映射和组相联映射。

当 Cache 存储器产生了一次访问失效之后，相应的数据应同时读入 CPU 和 Cache。但当 Cache 已存满数据后，新数据必须淘汰 Cache 中的某些旧数据。最常用的淘汰算法有随机淘汰法、先进先出法 (FIFO) 和近期最少使用淘汰法 (LRU)。

(2) 虚拟存储器

虚拟存储器是由主存、辅存、存储管理单元及操作系统中存储管理软件组成的存储系统，该存储系统对程序员的使用通常是透明的。可以使用的内存空间远远大于主存的物理空间，但实际上并不存在那么大的主存。故称其为虚拟存储器。

虚拟存储器分为页式虚拟存储器、段式虚拟存储器和段页式虚拟存储器。

(3) 磁盘阵列技术

磁盘阵列是由多台磁盘存储器组成的快速、大容量、高可靠的外存子系统。现在常见的磁盘阵列称为廉价冗余磁盘阵列 (RAID)。RAID 分为 RAID0, RAID1, ..., RAID10。

流水线技术

流水线技术其实是通过硬件的并行运作提高系统性能的常用方法，其基本思想在冯·诺依曼第一台存储程序计算机中就已经提出。

流水线技术的基本原理实际上是一种任务分解的技术。它把一件任务分解成若干个可以顺序执行的子任务，不同的子任务由不同的执行部件负责执行，而这些机构可以同时并行工作。在任一时刻，任一任务只占用其中一个执行机构，这样就可以实现多个任务的重叠执行，以提高工作效率。

流水线技术包括指令流水线和运算操作流水线。需要注意的是，对流水线技术而言，其对性能的提高程度取决于其执行顺序中最慢的子任务。

在实际情况中，流水线各个阶段可能会相互影响，阻塞流水线，使其性能下降。影响流水线性能的主要因素有两个：执行转移指令和共享资源冲突。

在实际处理中，为了使流水线能维持最大的吞吐率，同时确保流水

线各段不会产生冲突，就需要对流水线进行很好的控制。一般采用预留表的方式来预测冲突。预留表是从流水线设计直接推导出来的，表中列出的是流水线上各个部件操作的时间信息，每一行代表流水线中的一段，而每一列则代表一个时间步。

并行处理机

并行处理机有时也称为阵列处理机，其得名于在单一控制部件控制下的由多个处理单元所构成的阵列。主要用于那些需要进行大量高速向量或矩阵运算的应用领域。

并行处理机的并行性来源于资源重复，把大量相同的处理单元 PE 通过互连网络连接起来，在中央控制器 (CU) 控制下，对各自分配来的数据并行地完成同一条指令所规定的操作。PE 是不带指令控制部件的算术逻辑运算单元。

并行处理机的特点：

- ① 强大的向量运算能力。
- ② 并行方式。并行处理机的并行性来源于资源重复，而不是时间重叠，利用的是并行性中的同时性，而不是并发性。并行机提高运算速度的主要方法是增加处理单元的个数。
- ③ 适用于专门领域，如矩阵运算、向量运算等。
- ④ 标量运算速度对系统性能的影响也很明显。
- ⑤ 具有向量化功能的高级语言编译程序有助于提高并行处理机的通用性，以及减少编译时间。

多处理机

多处理机通常具有两个或两个以上的处理机，它们相互共享输入/输出子系统，在统一的操作系统控制下，通过共享主存或高速通信网络进行通信，协同求解一个较复杂的问题。多处理机通过利用多台处理机进行多任务处理来提高速度，利用系统的重组能力来提高可靠性、适应性和可用性。多处理机是新一代计算机结构的基本特征，多处理机包括共享存储器和分布存储器两种不同的存储结构。

多处理机属于 MIMD 系统，与 SIMD 的并行处理机相比，有很大的差别。其根源就在于两者的并行性层次不同，多处理机要实现的是更高层次的作业任务间的并行。

大规模并行处理机 (MPP) 是由众多的微处理器 (从几百到上万) 组成的大规模并行处理系统。MPP 可以采用市场上出售的 RISC 处理器，所以有很高的性价比。

计算机系统 可靠性

计算机系统的可靠性是指从它开始运行 ($t=0$) 到某时刻 t 这段时间内能正常运行的概率。

常见的系统可靠性数学模型有 3 种。

(1) 串联系统

假设一个系统由 N 个子系统组成，当且仅当所有的子系统都能正常

工作时，系统才能正常工作，这种系统称为串联系统。

设各子系统的可靠性为 R_1, R_2, \dots, R_n ，则整个串联系统的可靠性为：

$$R = 2 - (1 - R_1) \times (1 - R_2) \times \dots \times (1 - R_n)$$

设各子系统的失效率为 λ_i ，则整个串联系统的失效率为：

$$\lambda = \lambda_1 + \lambda_2 + \dots + \lambda_n$$

(2) 并联系统

假如一个系统由 N 个子系统组成，只要有一个子系统正常工作，系统就能正常工作，这样的系统称为并联系统。

设各子系统的可靠性为 R_1, R_2, \dots, R_n ，则整个并联系统的可靠性为：

$$R = R_1 \times R_2 \times \dots \times R_n$$

设各子系统的失效率为 λ ，则整个并联系统的失效率为：

$$\mu = \frac{1}{\frac{1}{\lambda} \sum_{j=1}^N \frac{1}{j}}$$

(3) N 模冗余系统

N 模冗余系统由 N 个 ($N > 2n$) 相同的子系统和一个表决器组成，表决器把 N 个子系统中占多数的相同结果的输出作为系统的输出。

设各子系统的可靠性均为 R_0 ，则整个 N 模冗余系统的可靠性为：

$$R = \sum_{j=n+1}^N \binom{j}{N} R_0^j (1 - R_0)^{N-j}$$

其中， $\binom{j}{N}$ 表示从 N 个元素中取 j 个元素的组合数。

操作系统类型和功能

根据使用环境和对用户作业的处理方式划分，操作系统的基本类型可以分为批处理操作系统、分析操作系统和实时操作系统等3大类型。

分时操作系统使多个用户同时以会话方式控制程序运行，每个用户都感到似乎各自有一台独立的、支持自己请求服务的系统。

实时系统往往是专用的，系统与应用很难分离，常常紧密结合在一起。实时系统并不强调资源利用率，而更关心及时性（时间紧迫性）、可靠性和完整性。实时系统又分成实时过程控制与实时信息处理两种。

网络环境下的操作系统又分成网络操作系统和分布式操作系统。分布式操作系统要求一个统一的操作系统，负责全系统的资源分配和调度，为用户提供统一的界面。它是逻辑上紧密耦合的系统。网络操作系统用户则需指明要使用哪一台计算机上的哪个资源。

操作系统主要有5个功能模块：处理器管理、存储管理、设备管理、文件管理和用户接口。

处理机管理

(1) 进程

进程是一个程序关于某个数据集的一次运行过程。也就是说，进程是运行中的程序，是程序的一次运行活动。相对于程序，进程是动态的

概念，而程序是静态的概念，是指令的集合，因而进程具有动态性和并发性。

在操作系统中，进程是进行系统资源分配、调度和管理的最小单位，注意，现代操作系统中还引入了线程（Thread）这一概念，它是处理器分配资源的最小单位。

(2) 进程的状态及其转换

多道系统中，进程的运行状态是不断变化的。它在处理器上的交替运行，使它的运行状态不断变化，最基本的状态有3种，即运行、就绪和阻塞。

- ① 运行：正占用处理器。
- ② 就绪：只要获得处理器即可运行。
- ③ 阻塞：正等待某个事件的发生。

(3) 进程控制块

进程是一个动态的概念，在操作系统中，引入数据结构-进程控制块（PCB）来描述进程。PCB是进程存在的唯一标志，PCB描述了进程的基本情况。从静态的观点看，进程由程序、数据和进程控制块组成；从动态的观点看，进程是计算机状态的有序集合。

程序是进程运行所对应的运行代码，一个进程对应于一个程序，一个程序可以同时对应于多个进程，这个程序代码在运行过程中不会改变，常称为纯码程序或可重入程序，它们是可共享的程序。

进程控制块保存进程状态、进程性质（如优先程度）、与进程有关的控制信息（如参数、信号量和消息等）、相应队列和现场保护区域等。进程控制块随着进程的建立而产生，随着进程的完成而撤销。

PCB是操作系统内核中最主要的数据结构之一，它既是进程存在的标志和调度的依据，又是进程可以被打断并能恢复运行的基础。操作系统核心通过PCB管理进程，一般PCB是常驻内存的，尤其是调度信息必须常驻于内存。

(4) 进程管理

在操作系统中有许多进程，它们对应着不同的或相同的程序，相互竞争地使用着系统的资源。进程管理涉及到进程控制、队列管理和进程调度等。

进程的生命过程从创建它时开始，直至任务终止而消亡，其间会经历各种状态的转换，它们都是在操作系统控制下完成的。操作系统提供了对进程的基本操作，也称为原语。这些原语包括创建原语、阻塞原语、终止原语、优先级原语和调度原语。

进程调度即处理器的调度，它的主要功能是确定在什么时候分派处理器，并确定分给哪一个进程。在分时系统中，一般有一个确定的时间单位（时间片）。当一进程用完一个时间单位时，就发生进程调度，即让正在运行的进程改变状态并转入就绪队列的队尾，再由调度原语将绪

队列的首进程取出，投入运行。

进程调度的方法基本上分为两类：非抢占式调度与抢占式调度。所谓非抢占式调度，是指一旦某个作业或进程占有了处理器，别的进程就不能把处理器从这个进程那里夺走；相反，如果别的进程可将处理器从这个进程手中夺走，则是抢占式调度。

进程调度的算法采用服务于系统目标的策略，对于不同的系统与系统目标，常采用不同的调度算法，如先来先服务、优先数调度和轮转法等。

(5) 管程

管程是一种并发性的结构，包括用于分配一个特定的共享资源或一组共享资源的数据和过程。为了完成分配资源的功能，进程必须调用特定的管程入口。许多进程可能打算在不同的时间进入管程，但在管程边界上严格地实施互斥，在某一时刻，只允许一个进程进入。当管程中已有一个进程时，其他希望进入管程的进程必须等待。这种等待是由管程自动管理的。

管程中的数据或者是管程中所有的全局变量，或者是某个特定过程的局部变量。所有这些数据只能在管程内访问，在管程外的进程无法访问管程内的数据，这叫做信息掩蔽。

存储管理

现代计算机系统中的存储系统常是多级存储体系，至少有主存（内存）和辅存（外存）两级，有的系统有更多级数。主存大小由系统硬件决定，是实实在在的存储，它的存储容量受到实际存储单元的限制。虚拟存储（简称虚存）不考虑实际主存的大小和数据存取的实际地址，只考虑相互有关的数据之间的相对位置，其容量由计算机的地址的位数决定。

设备管理

设备管理是对计算机输入/输出系统的管理。其主要任务有：实现对外部设备的分配和回收；启动外部设备；控制输入/输出设备与处理器或主存间交换数据；实现对磁盘的调度；处理设备的中断；实现虚拟设备等。

外部和主存之间常用的传输控制方式有4种：程序控制方式、中断方式、直接存储访问（DMA）方式和通道方式。

文件管理

(1) 文件系统

操作系统的文件系统包括两个方面：一方面包括负责管理文件的一组系统软件，另一方面包括被管理的对象文件。文件系统的主要目标是提高存储器的利用率，接受用户的委托，实施对文件的操作。主要问题是管理辅助存储器，实现文件从主存空间到辅存地址空间的转换，决定文件信息的存放位置、存放形式和存取权限，实现文件和目录的操作，提供文件共享能力和安全设施，提供友好的用户接口。

(2) 文件的结构和组织

文件的结构是指文件的组织形式。从用户观点所看到的文件组织形

式，称为文件的逻辑结构；从实现观点考查文件在辅助存储器上的存放方式，称为文件的物理结构。

文件的逻辑组织是为了方便用户使用。一般文件的逻辑结构可以分为两种：无结构的字符流文件和有结构的记录文件，后者也称为有格式文件。

优化文件的物理结构是为了提高存储器的利用效率和降低存取时间。文件的存储设备通常被划分为大小相同的物理块，物理块是分配和传输信息的基本单位。文件的物理结构是指文件在存储设备上的存储方法。文件的物理结构涉及文件存储设备的组块策略和文件分配策略，决定文件信息在存储设备上的存储位置。

作业管理和 用户接口

根据使用环境和对用户作业的处理方式划分，操作系统的基本类型可以分为批处理操作系统、分析操作系统和实时操作系统3大类型。

分时操作系统使多个用户同时以会话方式控制程序的运行，每个用户都感到似乎各自有一台独立的、支持自己请求服务的系统。

实时系统往往是专用的，系统与应用很难分离，常常紧密结合在一起。实时系统并不强调资源利用率，而更关心及时性（时间紧迫性）、可靠性和完整性。实时系统又分成实时过程控制与实时信息处理两种。

网络环境下的操作系统又分成网络操作系统和分布式操作系统。分布式操作系统要求统一的操作系统，负责全系统的资源分配和调度，为用户提供统一的界面。它是逻辑上紧密耦合的系统。而网络操作系统用户则需指明要使用哪一台计算机上的哪个资源。

操作系统主要有5个功能模块：处理器管理、存储管理、设备管理、文件管理和用户接口。

其他管理

(1) 死锁问题

如果一个进程正在等待不可能发生的事件，则称该进程处于死锁状态。系统发生死锁是指一个或多个进程处于死锁状态。产生死锁的主要原因是共享的系统资源不足，资源分配策略和进程的推进顺序不当。系统资源既可能是可重用的永久性资源，也可能是消耗性的临时资源。处于死锁状态的进程不能继续运行又占有了系统资源，阻碍其他进程的运行。对待死锁的策略主要有：

① 死锁的预防

也就是不让任何能够产生死锁的必要条件发生。

② 死锁的避免

不对用户进程的推进顺序加以限制，在进程申请资源时先判断这次分配是否安全，只执行安全的实施分配，典型的算法是银行家算法。

③ 死锁的检测

采用资源请求分配图的化简方法判断是否发生了不安全状态。资源请求分配图是一种有向图，表示进程与资源之间的关系。死锁的检测是在需要的时刻执行的，当发现系统处于不安全状态时，即执行死锁的解

除策略。

④ 死锁的解除

解除死锁的基本方法是剥夺。一种方法是把资源从一些进程处剥夺分给别的进程，被剥夺资源的进程则需回退到请求资源处重新等待执行；另一种方法是终止一个进程，剥夺其全部资源，以后再重新运行被终止的进程。

(2) 多重处理器系统与线程

多重处理系统的主要目标是为了提高系统的处理能力，也是为了提高系统可靠性。多重处理系统的操作系统除了具有单处理器操作系统的功能以外，还应提供处理器的负载平衡、处理器发生故障后的结构重组等功能。一般多重处理系统的操作系统可以分为主从式、分离执行式和移动执行式等3类。

对称多处理器系统SMP是由若干同构甚至相同的处理器构成的系统。Solaris 和Windows NT 等操作系统支持SMP系统。操作系统提供了线程(Thread)机制以发挥多个处理器的作用。在多线程系统中，一个进程可以由一个或多个线程构成。进程是资源分配的基本单位，也是被保护的基本单位。一个进程对应于一个保存进程映象的虚地址空间，每一线程可以独立运行，一个进程的线程共享这个进程的地址空间。有多种方法可以实现多线程系统，一种方法是核心级线程，另一种方法是用户级线程，也可以把两者组合起来。

常用操作系统

应用程序所需要的服务。应用程序组可以构成应用子系统，如UNIX系统的源代码控制系统(SCCS)、图形(X-Window、Motif)等。

Windows NT 系统是20世纪90年代的操作系统，适用于高档工作站平台、局域网服务器或者主干计算机。Windows NT 支持对称处理器结构，支持多线程并行，采用90年代操作系统技术(即微内核技术)，在体系结构上采用客户机/服务器模式。

1.2 典型题解

- 操作数所处的位置，可以决定指令的寻址方式。操作数包含在指令中，寻址方式为(1)；操作数在寄存器中，寻址方式为(2)；操作数的地址在寄存器中，寻址方式为(3)。
 - 立即寻址
 - 直接寻址
 - 寄存器寻址
 - 寄存器间接寻址
 - 立即寻址
 - 相对寻址
 - 寄存器寻址
 - 寄存器间接寻址
 - 相对寻址
 - 直接寻址
 - 寄存器寻址
 - 寄存器间接寻址

【解析】操作数作为指令的一部分而直接写在指令中，这种寻址方式称为立即数寻址方式，所以第(1)空的答案是 A。

指令所要的操作数已存储在寄存器中，或把目标操作数存入寄存器，把在指令中指出所使用寄存器的寻址方式称为寄存器寻址方式，所以第(2)空的答案是 C。操作数在存储器中，操作数的有效地址用 SI、DI、BX 和 BP 四个寄存器之一来指定，称这种寻址方式为寄存器间接寻址方式，所以第(3)空的答案是 D。指令所要的操作数存放在内存中，在指令中直接给出该操作数的有效地址，这种寻址方式为直接寻址方式。

【答案】(1) A (2) C (3) D

2. 某计算机的时钟频率为 400MHz，测试该计算机程序使用 4 种类型的指令。每种指令的数量及所需指令时钟数(CPI)如表 1-1 所示，则该计算机的指令平均时钟数约为_____。

表 1-1 每种指令的数量及所需指令时钟数

指令类型	指令数目 (条)	每条指令需时钟数
1	160000	1
2	30000	2
3	24000	4
4	16000	8

- A. 1.85 B. 1.93
C. 2.36 D. 3.75

【解析】由表 1-1 可知，总指令数为：

$$160000+30000+24000+16000=230000$$

总时钟数为

$$160000+30000 \times 2+24000 \times 4+16000 \times 8=444000$$

所以，指令平均时钟数为

$$444000/230000=1.93$$

所以本题的答案是 B。

【答案】B

3. 下面的描述中，_____不是 RISC 设计应遵循的设计原则。

- A. 指令条数应少一些
B. 寻址方式尽可能少
C. 采用变长指令，功能复杂的指令长度长而简单指令长度短
D. 设计尽可能多的通用寄存器

【解析】RISC 机为了避免访问内存占用过多时间，采用了大量寄存器，所以 D 选项说法是正确的；减少了访问内存指令、数据和指令缓存方法等，所以 A 和 B 选项说法是正确的；CISC 的特点是多采用变长指令，而 RISC 刚好相反，所以选项 C 的说法是错误的。

【答案】C

4. 计算机执行程序所需时间 P ，可用 $P=I \cdot CPI \cdot T$ 来估计，其中 I 是程序经编译后的机器指令数， CPI 是执行每条指令所需的平均机器周期数， T 为每个机器周期的时间。RISC 计算机是采用 (1) 来提高机器的速度。它的指令系统具有 (2) 的特点。指令控制部件的构建，(3)。RISC 机器又通过采用 (4) 来加快处理器的数据处理速度。RISC 的指令集使编译优化工作 (5)。
- (1) A. 虽增加 CPI ，但更减少 I B. 虽增加 CPI ，但更减少 T
 C. 虽增加 T ，但更减少 CPI D. 虽增加 I ，但更减少 CPI
- (2) A. 指令种类少 B. 指令种类多
 C. 指令寻址方式多 D. 指令功能复杂
- (3) A. CISC 更适于采用硬布线控制逻辑，而 RISC 更适于采用微程序控制
 B. CISC 更适于采用微程序控制，而 RISC 更知于采用硬布线控制逻辑
 C. CISC 和 RISC 都只采用微程序控制
 D. CISC 和 RISC 都只采用硬布线控制逻辑
- (4) A. 多寻址方式 B. 大容量内存
 C. 大量的寄存器 D. 更宽的数据总线
- (5) A. 更简单 B. 更复杂
 C. 不需要 D. 不可能

【解析】RISC 精简了指令系统，使指令种类更少，所以 (2) 选 A。但各指令所需要的机器周期数相差不大，而且每个机器周期所完成的操作也比较简单，因此没有必要再采用微程序设计，而是采用硬件布线的控制逻辑，使用了大量的寄存器，所以 (3) 选 B，(4) 选 C。这样，程序经过编译后指令数是增加了，但执行指令的平均机器周期减小了，使得程序的执行时间大大减少，所以 (1) 选 D。从编译器角度看，需要解决的问题复杂性没有降低，而指令的复杂度降低了，实际上把复杂性留给了编译系统，所以 (5) 选 A。

【答案】 (1) D (2) A (3) B (4) C (5) A

5. 减少指令执行周期数是 RISC 计算机性能提高的基础，它是通过 (1)、指令控制部件 (2) 代码、(3) 寄存器、(4) 寻址方式和限制访问内存来实现的。(5) 都是与 RISC 芯片有关的名字。
- (1) A. 超长指令字 (VLIW) B. 单指令多功能
 C. 精简指令系统 D. 适当扩充指令系统
- (2) A. 尽可能多用原有 B. 增加
 C. 软件固化 D. 不用
- (3) A. 增加 B. 减少
 C. 精选 D. 不用
- (4) A. 增加 B. 减少
 C. 精选 D. 不用
- (5) A. SPARC 和 MIPS B. SPARC 和 SCSI