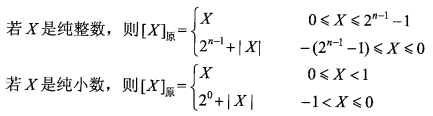
1.1.2.2

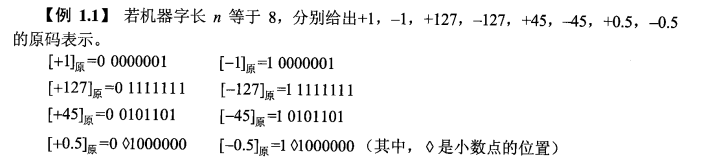
CPU由运算器、控制器、寄存器组、内部总线等部件组成

1.1.2.3

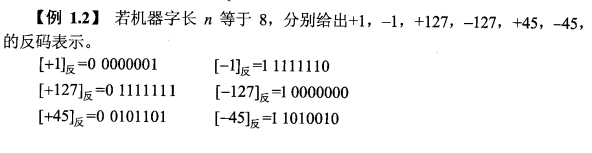
机器数：原码、反码、补码、移码

原码：数值X原码记为[X]原，机器字长记为n，





反码：正数反码与原码相同，负数反码是其绝对值按位求反（即原码的完全相反）



补码：正数的补码和原码相同，负数补码等于反码的末位+1，0为0 0000000，-0为0 0000000

移码：补码符号位取反

——浮点数、校验码-奇偶校验码

1.1.4.1奇偶校验：

1.1.4.2海明码：1/2/4/8/16位为校验码，分别计算对应位为1的所有值之和（或者说是进位）

1.2.1计算机体系结构

1.2.1.1

（1）计算机体系结构：计算机的概念性结构和功能属性

（2）计算机组织：计算机体系结构的逻辑实现，包括机器内的数据流和控制流的组成及逻辑设计等（即计算机组成原理）

（3）计算机实现：计算机组织的物理实现

1.2.1.2计算机体系结构分类

（1）按处理机数量分类：

单处理系统

并行处理与多处处理系统

分布式处理系统

1. 按微观上的并行程度分类

Flynn分类法

冯泽云分类法

Handler分类法

Kuck分类法

1.2.1.3指令系统

指令集体系结构的分类：

（1）操作数在CPU中的存储方式，即操作数从贮存中取出后保存的位置

（2）显式操作数的数量，即在典型的指令中有多少个显式命名的操作数

（3）操作数的位置，即任意一个ALU指令的操作数能否放在主存中，如何定位

（4）指令的操作，即在指令集中提供哪些操作

（5）操作数的类型与大小

按暂存机制分类，根据CPU内部存储操作数的区别分类

1. 堆栈
2. 累加器
3. 寄存器组

CISC和RISC

CISC：复杂指令集计算机

RISC：精简指令集周期

优化：指令集流水处理，思路同施工组织设计施工周期

1.2.2存储系统

CPU内部通用寄存器→Cache→主存储器→联机磁盘存储器→脱机光盘、磁盘存储器

1.2.2.1-2存储器层次结构

按位置分类：内存、外存

按材料分类：磁存储器（磁盘磁带等）、半导体存储器（根据元件等区分）、光存储器（光盘）

按工作方式分类：读写（RAM）、只读（ROM固定只读、PROM可编程的只读、EPROM可擦除可编程只读、EEPROM电擦除可编程只读、闪存即闪速存储器）

按访问方式分类：按地址访问、按内容访问

按寻址方式分类：随机存储器（RAM）、顺序存储器（SAM）、直接存储器（DAM）

1.2.2.3相联存储器

1.2.2.4高速缓存Cache：控制部分功能为判断CPU要访问的信息是否在Cache中，若在即为命中，直接对Cache寻址；未命中则按照替换原则决定调用主存内容

1.2.2.5虚拟存储器

1.2.2.6外存储器：即硬盘等大容量外部存储器

1.2.2.7-8磁盘阵列技术，存储域网络

1.2.3输入/输出技术

1.2.3.1微型计算机中最常用的内存与接口编址方法

1. 内存与接口地址独立编址方法：内存地址与接口地址完全独立，在编写读程序时易使用辨认
2. 统一编址方法：原则上用于内存的指令可以全部用于接口，指令上也不区分内存和接口，优势是增强对接口的操作功能，缺点是会导致内存地址不连续

1.2.3.2直接程序控制

1.2.3.3终端方式

1.2.3.4直接存储器存取方式

1.2.3.5输入输出处理机（IOP）

1.2.4总线结构

总线（Bus）指计算机设备与设备间传输信息的公共数据通道，是链接计算机硬件系统内多种设备的通信电路，由总线上所有设备共享

1.2.4.1总线分类：

数据总线：（Data Bus，DB）用于用于传送数据信息，双向

地址总线：（Address Bus）用于传送CPU发出的地址信息，单向，地址总线宽度决定了CPU最大寻址能力

控制总线：（Control Bus）用来传送控制信号、时序信号、状态信息等。其中每一条线的信息传送方向都是单项且确定的，作为整体则是双向

1.2.4.2常见总线：

ISA总线：工业标准总线，只能支持16位的IO设备，传输速率约为16Mb/s，也成为AT标准

EISA：32位

PCI：并行传输，目前使用广泛，适用于32位和64位，速率133（32）/266（64）

PCI Express：PCI-E，高频率高质量，根据总线位宽提升（X1 250Mb/s，X16 4G）

前端总线：链接CPU和北桥芯片的总线

RS-232C：一条串行外总线，所需传输线少

SCSI总线：小型计算机系统接口

SATA（Serial ATA）：主要用作主板和大量存储设备间的数据传输

USB：通用串行总线，由4调信号线组成，2跳数据2条电源，即插即用，支持热插拔

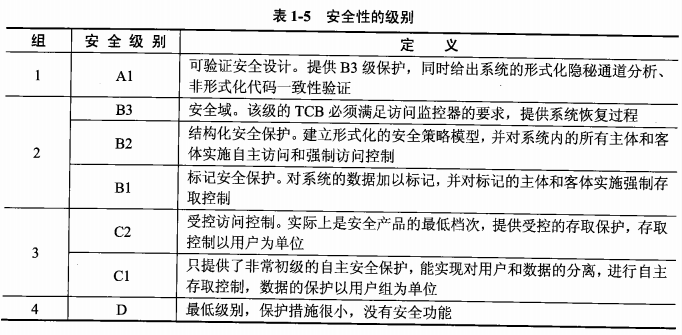
IEEE-1394：高速串行外总线

IEEE-488：并行总线接口标准总线

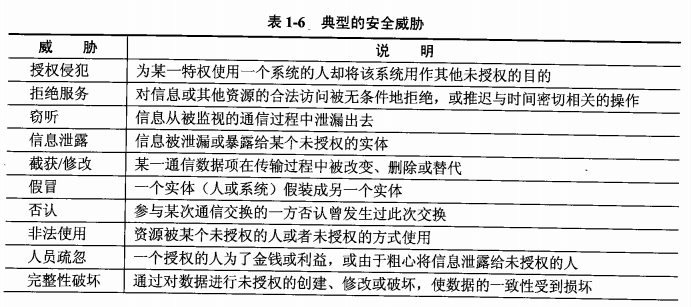
1.3安全性可靠性与系统性能评测

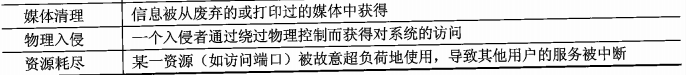
1.3.1计算机安全

1.3.1.1计算机安全等级



1.3.1.2安全威胁





1.3.1.3影响数据安全的因素：内部/外部

1.3.2加密与认证技术

1.3.2.1加密技术：对明文文件/数据按照算法进行处理，使其密文不可读

1对称加密（私钥）：

1. 数据加密标准算法（DES）：主要采用替换和移位
2. 三重DES（TDEA）：在DES的基础上使用，使用两个56位密钥K1、K2，发送使用K1加密、K2解密、K1加密
3. RC-5
4. 国际数据加密算法（IDEA）：类似三重DES，密钥长度128位
5. 高级加密标准（AES）：基于排列和置换

2非对称加密（公钥）：需要两个密钥：公钥和私钥，使用公钥加密的只能用对应的私钥解密，反之亦然

A生成一对密钥，并将其中一把作为公钥公开给其他方；得到公钥的B使用该密钥对信息加密后发给A；A再用另一把密钥对加密信息进行解密

3密钥管理：密钥包括证书和有效时间、已撤销密钥和证书的维护时间等

1. 密钥产生
2. 密钥的备份和恢复
3. 密钥更新
4. 多密钥管理

1.3.2.2认证技术

账户名/口令认证

使用摘要算法认证

基于PKI的认证：PKI是一种遵循既定标准的密钥管理平台，能够为所有网络应用提供加密和数字签名等密码服务及所必需的密钥和证书管理体系。包括：

1. 认证机构
2. 数字证书库
3. 密钥备份及恢复系统
4. 证书作废系统
5. 应用接口

SSL协议：安全套接层

1.3.3计算机可靠性

1. 计算机可靠性指从它开始运行t=0到某时刻t这段时间内能正常运行的概率，用R(t)表示。失效率指单位时间内失效的元件数与总元件数的比例
2. 计算机可靠性模型

1.3.4计算机系统的性能评价

1.3.4.1常用方法：

时钟频率

指令执行速度

等效指令速度法

速度处理速率

核心程序法

1.3.4.2基准测序程序（基准测序法）

整数测试程序、浮点数测试程序、SPEC基准程序

2.程序设计语言基础知识

2.1程序设计语言概述

2.1.1程序设计语言基本概念

2.1.1.1低级语言和高级语言

低级语言：机器语言和汇编语言

高级语言：面向各种应用的语言

2.1.1.2编译程序和解释程序

2.1.1.3程序设计语言的定义

语法、语义、语用

2.1.1.4程序设计语言分类

Fortran是第一个被广泛采用的高级语言

ALGOL 20世纪60年代

PASCAL衍生于ALGOL

C语言 兼顾高级语言和汇编语言的特点

C++ 与C兼容，多了封装、类和抽象，可以面向对象

C#（C Sharp）面向对象，运行于.NET

Objective-C根据C衍生

Java保留了C++的基本语法和面向对象，删掉了部分

Ruby解释型、面向对象、动态类型

PHP在服务器端执行，嵌入Html的脚本语言，混合了各种语法，支持几乎所有流行的数据库和操作系统

Python面向对象解释型，也是脚本语言，支持对操作系统的底层访问，提供了丰富的基本构建块，还可以用C、C++、Java等进行扩展

JavaScript脚本语言，为网页提供动态功能，通常嵌入在HTML中

Delphi可视化开发工具，在Windows上使用，对应的Linux产品是Kylix

Visual Basic.NET基于微软.NET

程序设计语言分类：

1. 命令式和结构化程序设计语言：计算看作动作的序列。一般用自顶向下逐步精化的方法编程、按模块组织的方法变成、程序只包含顺序、判定及循环构造且每种构造只允许单入口单出口
2. 面向对象程序设计语言：对象和类
3. 函数式程序设计语言：以演算为基础的语言。函数是一种对应规则，它使定义域每个元素和值域中唯一的元素相对应
4. 逻辑性程序设计语言：

2.1.2程序设计语言的基本成分

2.1.2.1数据成分

数据是程序操作的对象，具有存储类别、类型、名称、作用域和生存期等属性，使用时要分配内存空间。

1. 常量和变量
2. 全局量和局部量
3. 数据类型：基本类型、特殊类型、用户定义类型、构造类型、指针类型、抽象数据类型

2.1.2.2运算成分：指明允许使用的运算符号及规则

2.1.2.3控制成分：顺序结构、循环结构、选择结构、C++提供的部分结构

2.1.2.4传输成分

2.1.2.5函数：

函数定义：函数首部和函数体

返回值类型 函数名(形参表)

{

函数体;

}

函数声明：

返回值类型 函数名(参数类型表)

函数调用

函数名(实参表)

2.2语言处理程序基础

2.2.1汇编

2.2.1.1汇编是为特定的计算机设计的面向机器的符号化的程序设计语言。

汇编包含三类语句：

1. 指令语句：机器指令语句，能产生CPU直接识别执行的机器代码，在程序运行时完成
2. 伪指令语句：经汇编后不产生机器代码，操作在源程序被汇编时完成
3. 宏指令语句：允许用户将多次重复使用的程序段定义为宏

2.2.1.2汇编程序：将汇编语言编写的源程序翻译成机器指令程序，一般由于伪指令中后定义符号的存在需要扫描两次

2.2.2编译程序基本原理

2.2.2.1编译过程概述

源程序

→词法分析

从前往后扫描源程序，识别出单词符号

→语法分析

根据语言的语法规则将单词符号序列分解成各类语法单位

→语义分析

分析各语法结构的含义，检查源程序是否包含静态语义错误

→中间代码生成

→代码优化

→目标代码生成

→目标代码（符号表管理、出错处理）

2.2.2.2文法和语言的形式描述

2.2.2.3词法分析

2.2.2.4正则式与有限自动机之间的转换

2.2.2.5词法分析器的构造

2.2.2.6语法分析

2.2.2.7语法制导翻译和中间代码生成

2.2.2.8中间代码优化和目标代码生成

2.2.3解释程序基本原理

2.2.3.1解释程序的基本结构P110

3数据结构

3.1线性结构

特点为数据元素之间呈现线性关系

3.1.1线性表

3.1.1.1定义

一个线性表是n(n>0)个元素的有限序列，通常表示为(a1,a2...an)。

特点：

存在唯一一个“第一个”元素，其他元素有且只有一个直接前驱

存在唯一一个“最后一个”元素，其他元素有且只有一个直接后驱

3.1.1.2线性表的存储结构

顺序存储：一组地址连续的存储单元依次存储线性表中的数据元素

链式存储：用通过指针连接起来的节点来储存数据元素

3.1.2栈和队列

3.1.2.1栈

定义：栈是只能通过访问它的一端来实现数据存储和检索的一种线性数据结构，修改按照先进后出原则

3.1.2.2队列

定义：队列是先进先出，只允许在表的队尾插入元素和在队头删除元素。

3.1.3串

3.1.3.1定义：仅由字符构成的有限序列，如字符串

3.1.3.2串的存储结构：顺序、链式

3.1.3.3匹配模式：即子串（模式串）的定位操作

3.2数组、矩阵、广义表

3.2.1数组

3.2.1.1数组的定义及基本运算

定义：数组是定长线性表在维数上的扩展，即线性表中的元素又是一个线性表，n维数组是一种同构的数据结构，其每个元素类型相同、结构一致

3.2.1.2数组的顺序存储

数组一般不进行插入和删除，一旦定义，结构中的数据元素个数和元素之间的关系就不再变动

3.2.2矩阵

3.2.3广义表

线性表的推广，是由0或多个单元素或子表组成的有限序列，长度为元素个数，深度指展开后所含括号最大层数

3.3树

3.3.1树的定义

树中的一个数据元素可以有两个或者两个以上的直接后继元素

树是n个点的有限集合

树的定义是递归的

3.3.1.2基本概念

度：子树个数

父节点、子节点、叶子节点、内部节点

节点层次：根为1层，往下每层加1

树高度：最大层数

有序/无序

3.3.1.3二叉树

二叉树是空树、由一个根节点和两棵不相交且分别成为左右子树的二叉树构成。二叉树需要明确指明子树名称，最大度为2

3.3.2二叉树的性质与存储结构

3.3.2.1二叉树的性质

二叉树第i层最多有2的i-1次方个节点

高度为k的二叉树最多有2的k-1次方个节点

对于任意一棵二叉树，若其终端节点数为n0，度2的节点数为n2，则n0=n2+1

具有n个节点的完全二叉树的深度为(log2n)+1

3.3.2.2二叉树的存储结构

二叉树的顺序存储结构：从上往下遍历按顺序存储

3.3.3二叉树的遍历

3.3.4线索二叉树