

第 1 章 概 述

1.1 全国青少年信息学奥赛概述

全国青少年信息学奥林匹克系列活动由中国计算机协会(CCF)主办。该赛事旨在向在中学阶段的青少年普及计算机科学知识;给学校的信息技术教育课程提供动力和新的思路;给有才华的学生提供相互交流和学习的机会;通过竞赛和相关的活动培养和选拔优秀的计算机人才。

该赛事提出的背景:1984年,邓小平指出“计算机的普及要从娃娃做起”。中国计算机学会于1984年创办全国青少年信息学奥林匹克竞赛(NOI),这一新的活动形式受到党和政府的关怀,得到社会各界的关注与支持。从此每年举办一次NOI活动,吸引越来越多的青少年投身其中。多年来,通过竞赛活动培养和发现了大批计算机爱好者,选拔出了许多优秀的计算机后备人才。

为了在更高层次上推动普及,培养更多的计算机技术优秀人才,竞赛及相关活动遵循开放性原则,任何有条件和兴趣的学校和个人,都可以在业余时间自愿参加。NOI系列活动包括:全国青少年信息学奥林匹克竞赛、全国青少年信息学奥林匹克网上同步赛、全国青少年信息学奥林匹克联赛(National Olympiad in Informatics in Provinces,NOIP)、全国青少年信息学奥林匹克冬令营、全国青少年信息学奥林匹克选拔赛,以及国际信息学奥林匹克竞赛(International Olympiad in Informatics,IOI)。

1. NOI

自1984年至今,NOI是国内(包括港、澳在内)的省级代表队最高水平的大赛。每年经各省选拔产生5名选手(其中一名是女选手),由中国计算机学会在计算机普及度较高的城市组织进行比赛。这一竞赛记个人成绩,同时记团体总分。

NOI活动期间,同步举办夏令营和NOI网上同步赛,给那些程序设计爱好者和高手提供机会。为增加竞赛的竞争性、对抗性和趣味性以及可视化,NOI组织进行团体对抗赛,团体对抗赛实质上是程序对抗赛,其成绩纳入总分。

2. NOIP

NOIP自1995年至今,每年由中国计算机学会统一组织。NOIP在同一时间、不同地点以各省市为单位由特派员组织。全国统一大纲、统一试卷。初、高中或其他中等专业学校的学生可报名参加NOIP。NOIP分初赛和复赛两个阶段。初赛考查通用和实用的计算机科学知识,以笔试为主。复赛为程序设计,须在计算机上调试完成。参加初赛者须达到一定分数线后才有资格参加复赛。NOIP分普及组和提高组两个组别,难度不同,分别面向初中和

高中阶段的学生。

3. 冬令营

全国青少年信息学奥林匹克冬令营(简称冬令营)自 1995 年起,每年在寒假期间开展为期一周的培训活动。冬令营共 8 天,包括授课、讲座、讨论、测试等。参加冬令营的营员分正式营员和非正式营员。获得 NOI 前 20 名的选手和指导教师为正式营员,非正式营员限量自愿报名参加。冬令营的授课讲师是著名大学的资深教授及已获得国际金牌学生的指导教师。

4. APIO

亚洲与太平洋地区信息学奥赛(Asia Pacific Informatics Olympiad, APIO)于 2007 年创建,该竞赛为区域性的网上准同步赛,是亚洲和太平洋地区每年举办一次的国际性赛事,旨在给青少年提供更多的参加赛事的机会,推动亚太地区的信息学奥林匹克的发展。APIO 于每年 5 月举行,由不同的国家轮流主办。每个参赛团的参赛选手上限为 100 名,其中成绩排在前 6 名的选手作为代表该参赛团的正式选手统计成绩。APIO 中国赛区由中国计算机学会组织参赛,获奖比例参照 IOI。

5. 选拔赛

选拔参加 IOI 的中国代表队的竞赛(简称选拔赛)。IOI 的选手是从 NOI 中排名前 20 的选手中选拔出来的,排名前 4 的优胜者将代表中国参加国际竞赛。选拔依据为 NOI 成绩、冬令营成绩、论文和答辩成绩、平时的作业成绩、选拔赛成绩、口试成绩,上述各成绩加权产生最后成绩。

6. IOI

往年均由中国计算机学会组织代表队来代表中国参加国际每年举办一次的 IOI。中国是 IOI 创始国之一。出国参赛会得到中国科学技术协会和国家自然科学基金委员会的资助。自 1989 年开始,我国在 NOI、NOIP、冬令营、选拔赛的基础上,组织参加 IOI。

1.2 CCF 非专业级软件能力认证考试概述

CCF 非专业级软件能力认证(Certified Software Professional Junior/Senior, CSP-J/S)创办于 2019 年,是由 CCF 统一组织的评价计算机非专业人士算法和编程能力的活动。在同一时间、不同地点以各省市为单位由 CCF 授权的省认证组织单位和总负责人组织。全国统一大纲、统一认证题目,任何人均可报名参加。

CSP-J/S 分两个级别进行,分别为 CSP-J(入门级 Junior)和 CSP-S(提高级 Senior),两个级别难度不同,均涉及算法和编程。CSP-J/S 分第一轮和第二轮两个阶段。第一轮考查通用和实用的计算机科学知识,以笔试为主,部分省市以机试方式认证。第二轮为程序设计,须在计算机上调试完成。第一轮认证成绩优异者进入第二轮认证,第二轮认证结束后,CCF 将根据 CSP-J/S 各组的认证成绩和给定的分数线,颁发认证证书。CSP-J/S 成绩优异者可参加 NOI 省级选拔,省级选拔成绩优异者可参加 NOI。

1.3 全国青少年信息学奥赛考试大纲

全国青少年信息学奥林匹克联赛(NOIP)是全国青少年信息学奥林匹克竞赛(NOI)系列活动中的一个重要组成部分,旨在向中学生普及计算机基础知识,培养计算机科学和工程领域的后备人才。普及的重点是根据中学生的特点,培养学生学习计算机的兴趣,使得他们对信息技术的一些核心内容有更多的了解,提高他们创造性地运用程序设计知识解决实际问题的能力。

对学生的能力培养将注重以下的几方面。

- (1) 想象力与创造力。
- (2) 对问题的理解和分析能力。
- (3) 数学能力和逻辑思维能力。
- (4) 对客观问题和主观思维的口头和书面表达能力。
- (5) 人文精神,包括与人的沟通能力、团队精神与合作能力、恒心和毅力、审美能力等。

1. 竞赛形式

NOIP 分两个等级组:普及组和提高组。每组竞赛分两轮:初赛和复赛。

初赛形式为笔试,侧重考查学生的计算机基础知识和编程的基本能力,并对知识面的广度进行测试。初赛为资格测试,本省初赛成绩排在本赛区前 15% 的学生进入复赛。

复赛形式为上机编程,着重考查学生对问题的分析理解能力、数学抽象能力、编程语言的能力和编程技巧、想象力和创造性等。各省 NOIP 的奖项在复赛优胜者中产生。

比赛中使用的程序设计语言如下。

初赛: C++。

复赛: C++。

每年复赛结束后,各省必须在指定时间内将本省一等奖候选人的有关情况、源程序和可执行程序报送国家科学技术委员会。经复审和评测后,由中国计算机学会报送中国科学技术协会和教育部备案。中国计算机学会对各省获 NOIP 二等奖和三等奖的分数线或比例提出指导性意见,各省可按照成绩确定获奖名单。

2. 试题形式

每次 NOIP 的试题有 4 套:普及组初赛题 A1、普及组复赛题 A2、提高组初赛题 B1 和提高组复赛题 B2。其中,A1 和 B1 类型基本相同,A2 和 B2 类型基本相同,但题目不完全相同,提高组难度高于普及组。

初赛全部为笔试,满分为 100 分。试题由以下 4 部分组成。

- (1) 选择题:共 20 题,每题 1.5 分,共计 30 分。

每题有 5 个备选答案,前 10 题为单选题(即每题有且只有一个正确答案,选对得分),后 10 题为不定项选择题(即每题有 1~5 个正确答案,只有全部选对才得分)。普及组 20 个都是单选题。

- (2) 问题求解:共 2 题,每题 5 分,共计 10 分。

试题给出一个叙述较为简单的问题,要求学生对问题进行分析,找到一个合适的算法,并推算出问题的解。考生给出的答案与标准答案相同则得分;否则不得分。

(3) 程序阅读理解题:共4题,每题8分,共计32分。

题目给出一段程序(不一定有关于程序功能的说明),考生通过阅读理解该段程序给出程序的输出。输出与标准答案一致则得分;否则不得分。

(4) 程序完善题:共2题,每题14分,共计28分。

题目给出一段关于程序功能的文字说明,然后给出一段程序代码,在代码中略去了若干语句或语句的一部分,并在这些位置给出空格,要求考生根据程序的功能说明和代码的上下文填出被略去的语句。填对则得分;否则不得分。

复赛的题型和考试形式与NOI类似,全部为上机编程题,但难度比NOI低。题目包括4道题,每题100分,共计400分。每一试题包括:题目、问题描述、输入输出要求、样例描述及相关说明。测试时,测试程序为每道题提供了10~20组测试数据,考生程序每答对一组得10分,累计分即为该道题的得分。

其中普及组一场比赛共4道题,每题100分,共计400分,考试时间为3.5小时;提高组自2011年起由一试改为两试,分由两天进行。每场竞赛试题由原来的4题改为3题,每场考试时间为3.5小时。

从2016年开始,每年NOIP复赛普及组、提高组都将各有两题从NOI题库中选出。题目可能会变化,解法保持不变。

自2017年来,由于参赛人数增多,NOIP复赛规模的规则进行了调整,包括:每个省赛区可以设立多于两个的复赛考点(但必须在同一个城市),初赛进入复赛的比例和规模由各省赛区自行决定,在条件许可的情况下,鼓励更多选手参赛;复赛获奖比例将基本保持不变,全国一等奖获奖比例约为复赛参赛选手的20%。

3. 试题范围

(1) 初赛的考试内容与基本要求见表1-1。

表1-1 初赛的考试内容与基本要求

内 容	基 本 要 求
计算机的基本常识	(1) 计算机和信息社会(信息社会的主要特征、计算机的主要特征、数字通信网络的主要特征、数字化); (2) 信息输入输出的基本原理(信息交换环境、文字图形多媒体信息的输入输出方式); (3) 信息的表示与处理(信息编码、微处理部件 MPU、内存储结构、指令、程序和存储程序原理、程序的三种基本控制结构); (4) 信息的存储、组织与管理(存储介质、存储器结构、文件管理、数据库管理); (5) 信息系统组成及互联网的基本知识(计算机构成原理、槽和端口的部件间可扩展互连方式、层次式的互连结构、互联网络、TCP/IP、HTTP、Web 应用的主要方式和特点); (6) 人机交互界面的基本概念(窗口系统、人和计算机交流信息的途径(文本及交互操作)); (7) 信息技术的新发展、新特点、新应用等

内 容	基 本 要 求
计算机的基本操作	(1) Windows 和 Linux 的基本知识和常用操作； (2) 互联网的基本使用常识(网上浏览、搜索和查询等)； (3) 常用工具软件的使用(文字编辑、电子邮件收发等)
程序设计的基本知识	1. 数据结构 (1) 程序语言中的基本数据类型(字符、整数、长整、浮点)； (2) 浮点运算中的精度和数值比较； (3) 一维和二维数组(串)与线性表； (4) 结构类型(C++)。 2. 程序设计 (1) 结构化程序设计的基本概念； (2) 阅读理解程序的基本能力； (3) 具有将简单问题抽象成适合计算机解决的模型的基本能力； (4) 具有针对模型设计简单算法的基本能力； (5) 程序流程描述(自然语言/伪码/NS图/其他)； (6) 程序设计语言(C++)。 3. 基本算法处理 (1) 初等算法(计数、统计、数学运算等)； (2) 排序算法(冒泡法、插入排序、合并排序、快速排序)； (3) 查找(顺序查找、二分法)； (4) 回溯算法

(2) 复赛在初赛内容的基础上增加表 1-2 中的考试内容与基本要求。

表 1-2 复赛增加的考试内容与基本要求

内容	基 本 要 求
数据结构	(1) 指针类型； (2) 多维数组； (3) 线性表,包括栈、队列、单链表、双向链表及循环链表； (4) 树和图； (5) 文件操作(从文本文件中读入数据,并输出到文本文件中)
程序设计	(1) 算法的实现能力； (2) 程序调试的基本能力； (3) 设计测试数据的基本能力； (4) 程序的时间复杂度和空间复杂度的估计
数学与算法处理	(1) 离散数学和组合数学知识的应用(如排列组合、简单图论、数理逻辑)； (2) 分治思想； (3) 模拟法； (4) 贪心法； (5) 简单搜索算法(深度优先、广度优先)、搜索中的剪枝； (6) 动态规划的思想及基本算法

全国青少年信息学奥林匹克竞赛考试大纲会不定期进行修订,考试大纲请以中国计算机学会发布的最新版本大纲为准。

练 习 题

1. 【NOIP2018 普及组】中国计算机学会于()年创办全国青少年信息学奥林匹克竞赛。
A. 1983 B. 1984 C. 1985 D. 1986
2. 【NOIP2017 普及组】NOI 的中文意思是()。
A. 中国信息学联赛 B. 全国青少年信息学奥林匹克竞赛
C. 中国青少年信息学奥林匹克竞赛 D. 中国计算机协会
3. 【NOIP2005 普及组】下列活动中不属于信息学奥赛系列活动的是()。
A. NOIP B. NOI C. IOI D. 冬令营
E. 程序员等级考试

第 2 章 计算机基础

2.1 计算机的发展史

1. 第一台电子计算机的诞生

1946 年 2 月 14 日,世界上第一台通用电子数字计算机 ENIAC(Electronic Numerical Integrator and Computer)如图 2-1 所示,在美国宾夕法尼亚大学宣告研制成功。ENIAC 共使用了 18 000 个电子管、1500 个继电器及其他器件,占地面积约 170 平方米,重 30 余吨,计算速度是 5000 次/秒加法或 400 次/秒乘法。

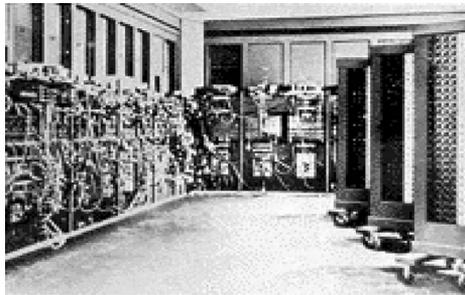


图 2-1 第一台通用电子数字计算机 ENIAC

2. 计算机发展的代次划分

第一代: 电子管计算机。

第二代: 晶体管计算机。

第三代: 中小规模集成电路计算机。

第四代: 大规模集成电路和超大规模集成电路计算机。

未来计算机: 智能计算机、神经网络计算机、生物计算机等。

3. 计算机的分类

(1) 按计算机的用途分类: 通用计算机和专用计算机。

(2) 按性能分类: 巨型机、大型机、中型机、小型机、微机。

(3) 按使用分类: 服务器、工作站、台式机、笔记本电脑、掌上电脑等。

4. 计算机的特点

(1) 处理速度快。

(2) 存储容量大,存储时间长久。

(3) 计算精确度高。

- (4) 具有逻辑判断能力。
- (5) 应用领域广泛。

5. 计算机的应用领域

计算机已经广泛地应用在人们的工作和生活中,主要的应用领域包括科学计算、信息处理、过程控制、计算机辅助系统、多媒体技术、计算机通信、人工智能等。

范例分析

1. 【NOIP2017 普及组】计算机应用的最早领域是(A)。
 - A. 数值计算 B. 人工智能 C. 机器人 D. 过程控制
2. 【NOIP2013 提高组】1948年,(D)将热力学中的熵引入信息通信领域,标志着信息论研究的开端。
 - A. 冯·诺依曼(John von Neumann) B. 图灵(Alan Turing)
 - C. 欧拉(Leonhard Euler) D. 克劳德·香农(Claude Shannon)
3. 【NOIP2012 提高组】目前计算机芯片(集成电路)制造的主要原料是(A),它是一种可以从沙子中提炼出的物质。
 - A. 硅 B. 铜 C. 锗 D. 铝
4. 【NOIP2011 普及组】摩尔定律(Moore's law)是由英特尔创始人之一戈登·摩尔(Gordon Moore)提出的。根据摩尔定律,在过去几十年以及在可预测的未来几年,单块集成电路的集成度大约每(C)个月翻一番。
 - A. 1 B. 6 C. 18 D. 36
5. 【NOIP2009 普及组】关于图灵机下面的说法中(D)是正确的。
 - A. 图灵机是世界上最早的电子计算机
 - B. 由于大量使用磁带操作,图灵机运行速度很慢
 - C. 图灵机是英国人图灵发明的,在二战中为破译德军的密码发挥了重要作用
 - D. 图灵机只是一个理论上的计算模型
6. 【NOIP2006 提高组】在下面各世界顶级的奖项中,为计算机科学与技术领域作出杰出贡献的科学家设立的奖项是(D)。
 - A. 沃尔夫奖 B. 诺贝尔奖 C. 菲尔兹奖 D. 图灵奖
 - E. 南丁格尔奖

2.2 计算机系统的组成

2.2.1 冯·诺依曼体系结构

美籍匈牙利数学家冯·诺依曼于1946年提出存储程序和程序控制原理,把程序本身当作数据来对待,程序和该程序处理的数据用同样的方式存储,并确定了存储程序计算机的五大组成部分和基本工作方法。他的理论为计算机的诞生和发展奠定了理论基础,直到今日,

我们使用的计算机仍然属于“冯·诺依曼机”。

冯·诺依曼理论的要点是：计算机的数制采用二进制；计算机应该按照程序顺序执行。他的这个理论被称为冯·诺依曼体系结构(见图 2-2)。

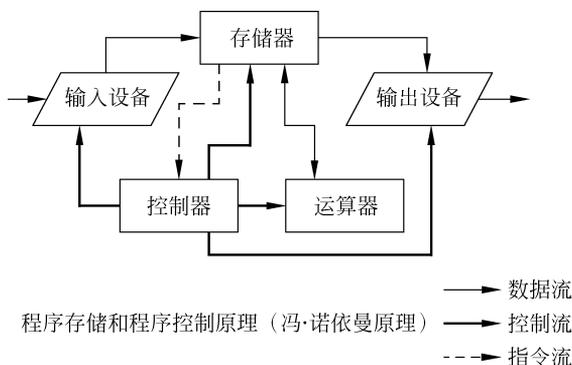


图 2-2 冯·诺依曼体系结构

总线是指计算机各部件进行信息传送和控制的公共通道,根据总线上传输数据的不同可以分以下几种。

(1) 数据总线(Data Bus, DBUS)用来在 CPU(Central Processing Unit, 中央处理器)与各部件之间传输数据信息,是双向传输的总线,传输方向由 CPU 控制。

(2) 地址总线(Address Bus, ABUS)用于传送 CPU 发出的地址信号,是一条单向传输总线,用于给出 CPU 所读取或发送的数据的存储单元地址或 I/O 设备地址。

(3) 控制总线(Control Bus, CBUS)用来传送控制信号、时序信号和状态信息等。其中有的是 CPU 向内存和外设发出的控制信号,有的则是内存或外设向 CPU 传递的状态信息。控制总线通过各种信号使计算机系统各个部件有条不紊地协调工作。

2.2.2 计算机系统架构与工作原理

一个完整的计算机系统由硬件系统和软件系统构成(见图 2-3)。没有软件系统的硬件称为“裸机”,“裸机”是无法直接使用的。

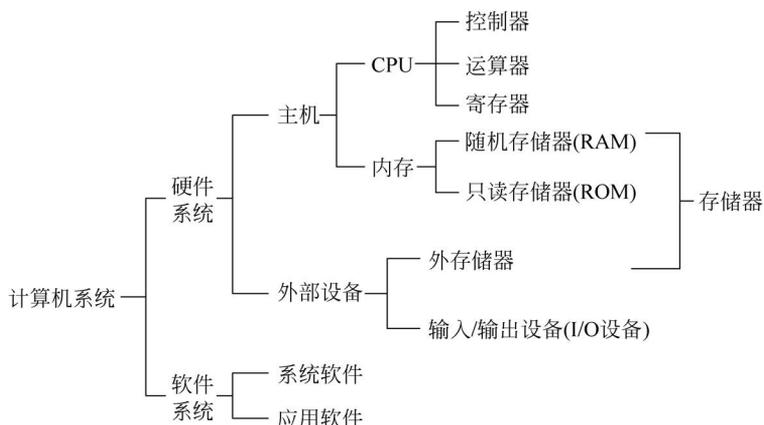


图 2-3 计算机系统架构

2.2.2.1 计算机硬件系统

根据冯·诺依曼体系结构,计算机硬件系统由五大部分组成:控制器、运算器、寄存器、输入设备和输出设备。

1. 中央处理器

1) 中央处理器由控制器、运算器和寄存器组成(见图 2-4)

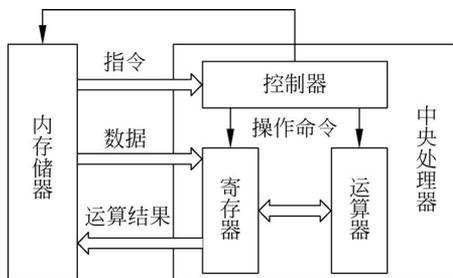


图 2-4 中央处理器的组成

控制器: 控制整个计算机的各个部件有条不紊地工作,它的基本功能是从内存取指令和执行指令。

运算器: 进行算术运算和逻辑运算。

寄存器: CPU 内部用来存放数据的一些小型存储区域,用来暂时存放参与运算的数据和运算结果。

2) 指令和指令系统

指令是一种采用二进制表示的命令语言,它用来规定计算机执行的操作及操作对象所在的位置。

指令系统则是指一个 CPU 能够执行的全部指令。

指令由操作码和操作数两部分组成。

(1) 操作码: 用来指明计算机应该执行何种操作的二进制代码。

(2) 操作数: 用来指明该操作处理的数据或数据所在存储单元的地址。

指令的执行过程大致如下。

(1) 指令预取: 指令预取部件从 Cache(高速缓冲存储器)或存储器中取得一条指令。

(2) 指令译码: 指令译码部件从指令预取部件获得指令,并对其进行分析。

(3) 获取操作数: 地址转换部件根据指令计算操作数的地址,并根据地址从 Cache 或存储器获取操作数。

(4) 运算: 运算器根据操作码的要求,对操作数完成指定操作。

(5) 保存: 若有必要,将结果保存到指定的寄存器或内存单元中。

(6) 修改指令地址: 为指令预取部件获取下一条指令做好准备。

计算机运行程序的过程就是一条一条执行指令的过程,程序中的指令和需要处理的数据都存放在存储器中,由 CPU 逐条取出并执行它所规定的操作。这就是存储程序和程序控制原理,也是到目前为止几乎所有计算机的基本工作原理。

3) CPU 的主要性能指标

计算机的运算速度是指计算机每秒执行的指令数,它是一项综合性的性能指标。其中 CPU 的主要性能指标包括主频和字长。

(1) CPU 的主频,即 CPU 内核工作的时钟频率。

(2) 字长又称数据宽度,一般由 1 个以上字节组成,是计算机进行数据处理时一次存取、加工和传送的数据长度。

2. 存储器

存储器主要用来存储程序及相关数据信息,存储器分为内存储器和外存储器。

1) 内存储器

内存储器简称内存,又称主存。内存在计算机中用来存放当前正在执行的程序和数据,内存和 CPU 构成了计算机的主机部分。内存储器是 CPU 可以直接存取数据的存储器,也是计算机程序运行过程中数据存储的最主要的场所,所有数据都通过内存储器和 CPU 进行交换,因此它的性能高低直接影响系统整体性能的发挥。

内存中包含很多存储单元,每个单元可以存放 1 个 8 位的二进制数,也就是 1 字节(Byte, B)。内存中的每个存储单元都有 1 个编号,也就是地址。CPU 存取内存数据时是按照地址进行的,通过地址编号寻找在存储器中的数据单元称为“寻址”。

存储器的容量就是指存储器中包含的字节总数。存储器容量的常用单位有 KB、MB、GB、TB 及 PB 等,它们之间的转换关系如下。

$1\text{KB}=1024\text{B}(1024=2^{10})$ 、 $1\text{MB}=1024\text{KB}$ 、 $1\text{GB}=1024\text{MB}$ 、 $1\text{TB}=1024\text{GB}$ 、 $1\text{PB}=1024\text{TB}$

内存储器通常分为以下三类。

(1) RAM(随机存储器)。用户程序和数据使用的存储器,断电后,RAM 中的信息随之丢失。

(2) ROM(只读存储器)。用来存放固定的程序和消息,断电后,ROM 中的信息保持不变。

(3) Cache(高速缓冲存储器)。在 CPU 和内存之间设置的高速小容量存储器,容量比较小但速度比主存高得多。CPU 处理数据时先从外存读入 RAM,再由 RAM 读入 Cache,然后 CPU 直接读取 Cache 进行处理。

2) 外存储器

外存储器又称为辅助存储器,相对内存储器来说,它的容量比较大,同时速度也相对较慢。常见的外存储器有软盘、硬盘、闪存和光盘等。

(1) 软盘。软盘是可以移动的存储介质,软盘的读写是通过软盘驱动器完成的,软盘驱动器设计能接收可移动式软盘。软盘存取速度慢,容量也小,但可装可卸,携带方便。

(2) 硬盘。硬盘分为机械式硬盘和固态硬盘(Solid State Disk 或 Solid State Driver, SSD)。其中机械式硬盘是以磁盘为存储介质的存储器,它是利用磁记录技术在涂有磁记录介质的旋转圆盘上进行数据存储的辅助存储器。固态硬盘,又称固态驱动器,是用固态电子存储芯片阵列制成的硬盘。硬盘存储器具有存储容量大、数据传输率高、存储数据可长期保存等特点。

(3) 闪存。闪存是一种特殊的半导体存储设备,一般应用于存储卡和 U 盘。闪存中的数据可以长期保存,即使断电数据也不会丢失。闪存的存取速度介于硬盘和其他外部存储器之间。

(4) 光盘。光盘是以光信息作为存储的载体并用来存储数据的存储设备,可以长期存放各种数字信息。光盘只是一个统称,主要分成两类:一类是只读型光盘;另一类是可记录型光盘,包括一次写入型和可重复擦写型。光盘的读写是通过光盘驱动器完成的,其存储容量大,存取速度较快。

3. 输入设备和输出设备

输入和输出的概念是相对于“主机”(CPU 和内存)来说的,数据进入主机则为输入,数据从主机向外传送则为输出。

输入设备是向计算机输入数据和信息的设备。最常见输入设备包括键盘和鼠标,其他如扫描仪、手写笔、麦克风等也都是输入设备。

输出设备是用于把计算机的各种计算结果数据或信息以数字、字符、图像、声音等形式呈现出来的设备。常见的输出设备有显示器和打印机,其他如绘图仪等也是输出设备。

磁盘和磁带等设备既可以向主机输入数据,也可以将主机的数据输出存储,因此这些设备既是输入设备也是输出设备。

2.2.2.2 计算机的主要性能指标

1. 运算速度

计算机的运算速度是衡量计算机性能的一项重要指标,通常所说的计算机运算速度 MIPS 指每秒处理的百万级的机器语言指令数,是一项综合性的性能指标。

(1) 主频。主频即 CPU 内核工作的时钟频率,主频的快慢很大程度决定了计算机的运算速度。微机一般采用主频来描述运算速度,主频越高,运算速度就越快。主频的单位一般用 GHz。

(2) 字长。字长又称数据宽度,是计算机进行数据处理时一次存取、加工和传送的数据宽度。字长直接反映了一台计算机的计算精度,同时,字长越大的计算机处理数据的速度就越快。现在常用的字长有 32 位、64 位。

2. 内存储器的性能

(1) 内存存取速度,即每次与 CPU 间处理单位数据的快慢。

(2) 内存容量,反映了计算机即时存储信息的能力,内存容量越大,系统功能就越强大,能处理的数据量就越庞大。

3. 输入/输出设备(I/O 设备)的速度

I/O 设备的速度指 CPU 与外部设备进行数据交换的速度。随着 CPU 主频速度的提升,存储器容量的扩大,系统性能的瓶颈越来越多地体现在 I/O 设备的速度上。主机 I/O 的速度取决于 I/O 总线的设计。

2.2.2.3 计算机软件系统

没有安装软件的计算机称为“裸机”,裸机是不能工作的。硬件系统提供了软件运行的平台。计算机软件系统(见图 2-5)主要分为系统软件和应用软件两大类。

1. 系统软件

系统软件负责管理、控制、维护、开发计算机软、硬件资源,提供给用户一个便利的操作界面和提供编制应用程序的资源环境。

1) 操作系统

操作系统(OS)位于整个软件的核心位置,其他系统软件处于操作系统的外层,应用软件则处于计算机软件的最外层。操作系统是计算机系统中最重要系统软件,它的主要功能是负责管理计算机系统硬件资源和软件资源,提高资源利用率,同时为计算机用户提供各种强有力的使用功能和方便的服务界面。只有在操作系统的支持下,计算机系统才能正常运行,如果操作系统遭到破坏,计算机系统就无法正常工作。

操作系统的功能包括:

(1) 处理机管理(主要控制和管理 CPU 的工作)。

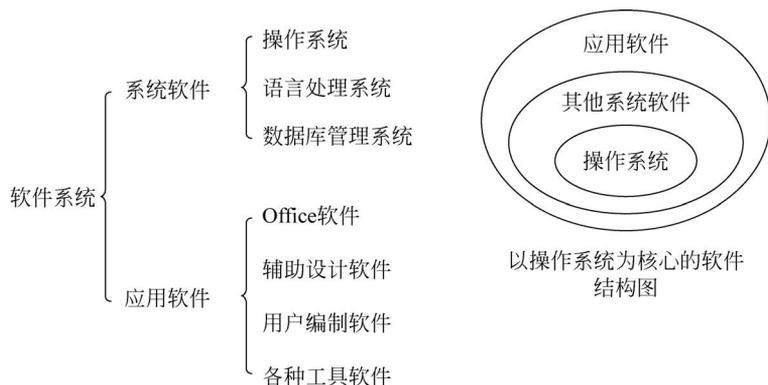


图 2-5 计算机软件系统

- (2) 存储管理(主要管理内存分配、保护、回收、扩充和地址映射等)。
- (3) 文件管理(支持文件的存储、检索、修改等操作,解决文件的共享、保密和保护)。
- (4) 设备管理(主要进行所有 I/O 设备的管理)。
- (5) 进程管理(也称作业管理,主要进行作业调度和作业控制等)。

常见的操作系统包括 Windows 系列、UNIX 系列、Linux 系列、macOS 系列等。

2) 程序设计语言及其处理程序

计算机程序是指一组指示计算机完成特定工作的指令,通常用某种程序设计语言编写,运行于计算机上。程序设计语言通常分为三类:机器语言、汇编语言、高级语言。

(1) 机器语言。

机器语言就是由“0”和“1”组成的二进制语言,是计算机唯一能直接识别、直接执行的计算机语言。机器语言依赖于计算机指令系统,不同型号的计算机,其机器语言是不同的,因此存在兼容性问题;机器语言的执行效率高,但是不便于记忆和理解,编写的程序难以修改和维护,因此很少直接使用机器语言编写程序。

(2) 汇编语言。

汇编语言是机器语言的进化,它用助记符来表示机器语言中的指令和数据,每一条汇编语言的指令就对应一条机器语言的代码。汇编语言编写的程序不能直接由计算机执行,必须先转换成计算机能直接识别的二进制代码才能执行,汇编语言的执行过程如图 2-6 所示。

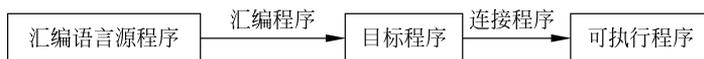


图 2-6 汇编语言的执行过程

汇编语言比机器语言简单,但是汇编语言和机器语言都是面向机器的程序设计语言,仍然属于低级语言,与运行的计算机的指令系统相关,程序的可移植性差。

(3) 高级语言。

高级语言是一种与硬件结构及指令系统无关,表达方式比较接近自然语言和数学表达式的一种计算机程序设计语言。高级语言描述问题的能力,通用性、可读性、可维护性都较好;但是高级语言的源程序不能被计算机直接识别,仍然需要转换。转换的方式有两种,分别为编译和解释(见图 2-7)。

编译方式是利用编译程序将高级语言源程序整个编译成等价的、独立的目标程序,然后

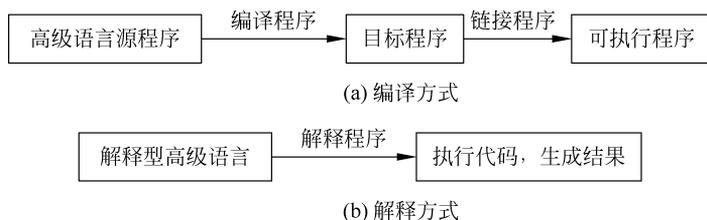


图 2-7 高级语言执行过程

通过链接程序将目标程序链接成可执行程序。

解释方式是将解释型高级语言通过解释程序逐句翻译,翻译一句执行一句,边翻译边执行,不产生目标程序。整个执行过程,解释程序都一直在内存中。

常见的高级语言有 C/C++、Java、C#、Pascal、Basic 等。

2. 应用软件

应用软件是为了满足用户不同领域、不同问题的应用需求而专门编制的软件。系统软件为应用软件提供了基础和平台,而应用软件通过系统软件的支持可以拓宽计算机系统的应用领域,扩展计算机的功能。

2.2.3 案例分析

1. 【CSP-S2019】编译器的功能是(B)。

- A. 将源程序重新组合
- B. 将一种语言(通常是高级语言)翻译成另一种语言(通常是低级语言)
- C. 将低级语言翻译成高级语言
- D. 将一种编程语言翻译成自然语言

【分析】 机器语言是能被计算机直接识别和运行的语言,高级语言源程序需要经过转换才能被计算机运行。转换的方式有编译和解释。编译器就起到这个转换的作用。

2. 【CSP-J2019】一个 32 位整型变量占用(C)字节。

- A. 32
- B. 128
- C. 4
- D. 8

【分析】 1 字节代表 8 个二进制位(bit),32 位占用 4 字节。

3. 【NOIP2018 普及组】以下(D)设备属于输出设备。

- A. 扫描仪
- B. 键盘
- C. 鼠标
- D. 打印机

【分析】 输入和输出都是相对于主机(CPU+内存)而言的,数据传向主机的称为输入,反之则为输出。因此扫描仪、键盘和鼠标均属于输入设备。

4. 【NOIP2018 普及组】1MB 等于(D)。

- A. 1000 字节
- B. 1024 字节
- C. 1000×1000 字节
- D. 1024×1024 字节

【分析】 存储器容量常用单位之间的转换关系为 $1\text{KB}=1024\text{Byte}(1024=2^{10})$, $1\text{MB}=1024\text{KB}$, $1\text{GB}=1024\text{MB}$, $1\text{TB}=1024\text{GB}$ 。

5. 【NOIP2018 提高组】下列属于解释执行的程序设计语言是(D)。

- A. C
- B. C++
- C. Pascal
- D. Python

【分析】 常见的解释执行的程序设计语言包括：Python、Ruby、JavaScript、Perl、PHP等。常见的编译执行的程序设计语言包括：C、C++、Java、Go、Swift等。

6. **【NOIP2017 普及组】**下列不属于面向对象程序设计语言的是(A)。

- A. C B. C++ C. Java D. C#

【分析】 常见的非面向对象程序设计语言包括 C、FORTRAN、COBOL、Pascal、Ada等；常见的面向对象程序设计语言包括 Java、C++、Python、Ruby、Swift等。

7. **【NOIP2016 普及组】**以下不是微软公司出品的是(D)。

- A. PowerPoint B. Word C. Excel D. Acrobat Reader

【分析】 PowerPoint、Word 和 Excel 属于微软公司 Office 软件套件中的软件，Acrobat Reader 是 Adobe 公司的软件产品。

8. **【NOIP2015 提高组】**下列说法中正确的是(A)。

- A. CPU 的主要任务是执行数据运算和程序控制
B. 存储器具有记忆能力，其中的信息任何时候都不会丢失
C. 两个显示器的屏幕尺寸相同，则它们的分辨率必定相同
D. 个人用户只能使用 WiFi 的方式连接到 Internet

【分析】 CPU 中包含运算器和控制器，用于控制整个计算机的各个部件有条不紊地工作，并进行算术运算和逻辑运算，故 A 选项正确。随机存储器(内存)断电以后其中的信息就消失了，故 B 选项错误。屏幕分辨率是指屏幕可显示的最高像素数目，屏幕尺寸是指屏幕面积，关于显示器还有一个术语叫“点距”，就是屏幕上像素与像素之间的距离，也就是代表单位面积内像素点数的一个值。屏幕尺寸和点距都一定时，屏幕的分辨率才一定。当两项中有一项发生变化，那么分辨率就会发生变化，故 C 选项错误。用户连接到 Internet 的方式有很多种，如 ADSL、FTTX+LAN 等方式。

9. **【NOIP2015 普及组】**操作系统的作用是(C)。

- A. 把源程序译成目标程序 B. 便于进行数据管理
C. 控制和管理系统资源 D. 实现硬件之间的连接

【分析】 操作系统是计算机系统最重要的系统软件，主要功能是负责管理计算机系统硬件资源和软件资源，提高资源利用率，同时为计算机用户提供各种强有力的使用功能和方便的服务界面。

10. **【NOIP2012 提高组】**目前个人计算机的(B)市场占有率最靠前的厂商包括 Intel、AMD 等公司。

- A. 显示器 B. CPU C. 内存 D. 鼠标

【分析】 目前 CPU 市场占有率最高的公司是英特尔(Intel)。英特尔是全球领先的半导体制造商之一，其 CPU 市场占有率高达 80% 以上。除英特尔之外，另一家全球知名的 CPU 制造商是 AMD(Advanced Micro Devices)，其在 CPU 市场上的份额正在逐渐增加。此外，还有一些其他的 CPU 制造商，如 ARM、IBM、NVIDIA 等，它们在特定领域有着较高的市场份额和影响力。

11. **【NOIP2012 提高组】**寄存器是(D)的重要组成部分。

- A. 硬盘 B. 高速缓存
C. 内存 D. 中央处理器(CPU)

【分析】 对于台式计算机的正常运行,以下硬件是必需的:①主板(Motherboard);②中央处理器(Central Processing Unit,CPU);③内存(Memory);④硬盘(Hard Disk Drive,HDD);⑤显卡(Graphics Card);⑥电源供应器(Power Supply Unit,PSU)。

17. **【NOIP2003 提高组】**下列计算机设备中既是输入设备,又是输出设备的是(B)。

- A. 键盘 B. 触摸屏 C. 扫描仪 D. 投影仪
E. 数字化仪

【分析】 键盘、扫描仪、数字化仪是输入设备,投影仪是输出设备,而触摸屏既是输入设备也是输出设备。

2.3 数制的概念及相互转换

计算机的基本功能是对数据进行运算和加工处理。计算机中的数据有两种:数值数据和非数值数据(信息),所有数据在计算机中都是用二进制数码表示的。

在计算机科学技术中,除了二进制之外,常用的还有十进制、八进制和十六进制,这些数制都是利用固定的数字符号和统一的规则来计数的方法,也称为进位记数制。

2.3.1 数制的术语

一种进位记数制包含一组数码符号和三个基本因素。

(1) 数码:一组用来表示某种数制的符号。例如,十六进制的数码是0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F;二进制的数码是0、1。

(2) 基数:指该数制可使用的数码个数。例如,十进制的基数是10;二进制的基数是2。

(3) 数位:数码在一个数中所处的位置。

(4) 权:权是基数的幂,表示数码在不同位置上的数值。

2.3.2 常用的记数制

1. 十进制

十进制是我们日常生活中经常使用的数制。十进制记数制的数码包括0、1、2、3、4、5、6、7、8、9,基数为10,位权为 10^n (n 为符号所处的数位)。每个数位计满10向高位进一,即“逢十进一”。

2. 二进制

二进制是计算机中使用的数制。二进制记数制的数码包括0、1,基数为2,位权为 2^n (n 为符号所处的数位)。每个数位计满2向高位进一,即“逢二进一”。

3. 八进制

八进制记数制的数码包括0、1、2、3、4、5、6、7,基数为8,位权为 8^n (n 为符号所处的数位)。每个数位计满8向高位进一,即“逢八进一”。

4. 十六进制

十六进制记数制的数码包括 0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F，基数为 16，位权为 16^n (n 为符号所处的数位)。每个数位计满 16 向高位进一，即“逢十六进一”。

2.3.3 数制间的相互转换

不同数制的数据有三种书写方式。

(1) $11101101_{(2)}$, $331_{(8)}$, $35.81_{(10)}$, $FA5_{(16)}$ 。

(2) $(10110.011)_2$, $(755)_8$, $(139)_{10}$, $(AD6)_{16}$ 。

(3) 10101001B, 789O, 3762D, 2CE6H (其中的 B、O、D、H 分别表示二进制(Binary)、八进制(Octal)、十进制(Decimal)和十六进制(Hexadecimal))。表 2-1 给出了数据在各种进制制中的表示示例。

表 2-1 数制间的转换

十进制(D)	二进制(B)	八进制(O)	十六进制(H)
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A 或 a
11	1011	13	B 或 b
12	1100	14	C 或 c
13	1101	15	D 或 d
14	1110	16	E 或 e
15	1111	17	F 或 f

1. 二进制数与十进制数的转换

1) 二进制数转十进制数

二进制数转十进制数采用“按权展开”的方式。

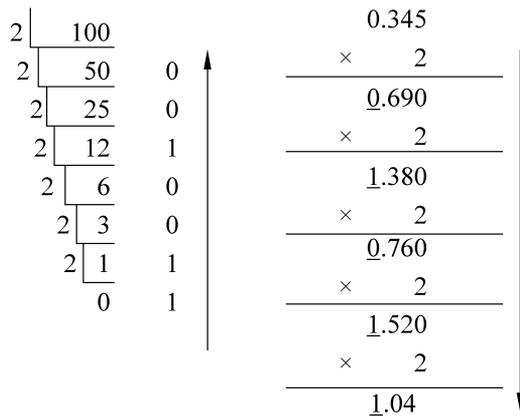
例: $(1011.101)_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} = 8 + 0 + 2 + 1 + 1/2 + 0 + 1/8 = (11.625)_{10}$

2) 十进制数转二进制数

整数部分: 除以 2 取余数, 直到商为 0, 余数从右到左排列。

小数部分: 乘以 2 取整数, 整数从左到右排列。

例: $100.345_{(10)} = 1100100.01011_{(2)}$



一个有限的十进制小数并非一定能够转换成一个有限的二进制小数,即上述过程中乘积的小数部分可能永远不等于0,可按要求进行到某一精确度为止。

2. 八进制数与十进制数的转换

1) 八进制数转十进制数

同样采用“按权展开”的方法。

例: $(2576)_8 = 2 \times 8^3 + 5 \times 8^2 + 7 \times 8^1 + 6 \times 8^0 = (1406)_{10}$

2) 十进制数转八进制数

采用与十进制数转二进制数相同的方法。

例: $100_{(10)} = 144_{(8)}$

$$\begin{array}{r|l} 8 & 100 \\ \hline 8 & 12 \quad 4 \\ 8 & 1 \quad 4 \\ & 0 \quad 1 \end{array}$$

3. 十六进制数与十进制数的转换

1) 十六进制数转十进制数

同样采用“按权展开”的方法,十六进制非数值的数码先转换为对应的十进制数,如 A 转换为 10, F 转换为 15。

例: $(F.B)_{16} = 15 \times 16^0 + 11 \times 16^{-1} = 15 + 11/16 = (15.6875)_{10}$

2) 十进制数转十六进制数

采用与十进制数转二进制数相同的方法。

例: $100_{(10)} = 64_{(16)}$

$$\begin{array}{r|l} 16 & 100 \\ \hline 16 & 6 \quad 4 \\ & 0 \quad 6 \end{array}$$

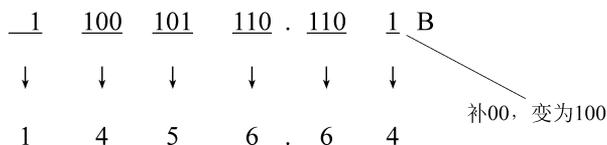
4. 二进制数与八进制数的转换

1) 二进制数转八进制数

二进制数转换成八进制数的方法是:将二进制数从小数点开始,整数部分从右向左 3

位一组,小数部分从左向右3位一组,若不足三位用0补足即可。

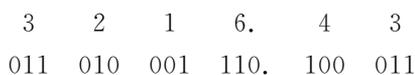
例: $(1100101110.1101)_2 = (1456.64)_8$



2) 八进制数转二进制数

八进制数转换成二进制数的方法是:以小数点为界,向左或向右每一位八进制数用相应的3位二进制数取代,然后将其连在一起即可。若中间位不足3位,在前面用0补足。

例: $(3216.43)_8 = (11010001110.100011)_2$

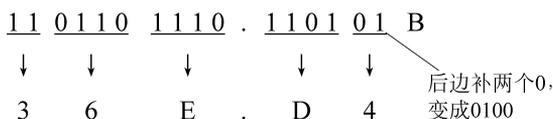


5. 二进制数与十六进制数的转换

1) 二进制数转十六进制数

二进制数转换成十六进制数的方法是:从小数点开始,整数部分从右向左4位一组;小数部分从左向右4位一组,不足4位用0补足,每组对应一位十六进制数即可得到十六进制数。

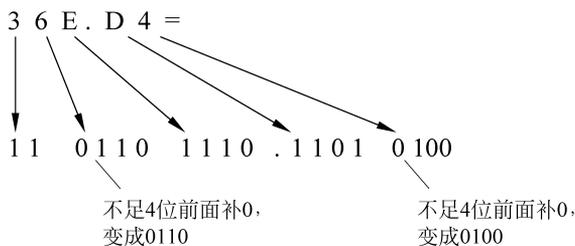
例: $(1101101110.110101)_2 = (36E.D4)_{16}$



2) 十六进制数转二进制数

十六进制数转换成二进制数的方法是:以小数点为界,向左或向右每一位十六进制数用相应的4位二进制数取代,然后将其连在一起即可。

例: $(36E.D4H)_{16} = (1101101110.110101)_2$



6. 八进制数与十六进制数的转换

八进制数与十六进制数之间的转换,一般通过二进制数作为桥梁,即先将八进制或十六进制数转换为二进制数,再将二进制数转换成十六进制数或八进制数。

例: $(637.15)_8 = (19F.34)_{16}$