教育部大学计算机课程改革项目规划教材

程序设计思想与方法
——问题求解中的计算思维

陆朝俊　编著

高等教育出版社
HIGHER EDUCATION PRESS

教育部大学计算机课程改革项目规划教材

程序设计思想与方法

——问题求解中的计算思维

Chengxu Sheji Sixiang yu Fangfa

陆朝俊 编著

高等教育出版社·北京
HIGHER EDUCATION PRESS BEIJING

内容提要

当前高等学校计算机基础教学的一个改革和发展方向是将“计算思维”作为程序设计课程主线，让学生学会如何像计算机科学家那样思考并解决问题，这与传统的以一门编程语言为课程核心的做法是完全不同的。

本书是作者参与教育部大学计算机课程改革项目而形成的产物，全书以问题求解中的计算思维为线索，介绍一般的程序设计思想与方法。具体内容包括：信息和信息处理过程的表示，处理流程的结构化和模块化设计方法，从面向过程方法到面向对象方法的发展，图形和 GUI 编程，事件驱动编程和并发编程，蒙特卡罗模拟方法设计等。要强调的是，以上所有思想和方法都是通过生动的实例演示而非枯燥的原理灌输而展开介绍的。

虽然本书不是关于编程语言的教材，但是通过本书的学习，学生可以掌握简单而强大的 Python 语言，能够利用 Python 实现自己对实际问题解决方法的思考和设计。

本书的目标受众是高等学校中非计算机科学与技术专业的学生，当然也适用于计算机科学与技术专业学生的入门课程。实际上，由于本书编写时注重“手把手似的”循序渐进教学，任何具备高中文化水平的人都可以通过自学掌握本书内容。

图书在版编目（CIP）数据

程序设计思想与方法：问题求解中的计算思维 / 陆朝俊编著. --北京：高等教育出版社，2013.11

ISBN 978-7-04-038569-4

Ⅰ. ①程… Ⅱ. ①陆… Ⅲ. ①程序设计－高等学校－教材 Ⅳ. ①TP311.1

中国版本图书馆 CIP 数据核字（2013）第 29602 号

策划编辑 耿 芳　责任编辑 张海波　封面设计 于文燕　版式设计 王艳红
插图绘制 尹 莉　责任校对 刘春萍　责任印制 朱学忠

出版发行 高等教育出版社　咨询电话 400-810-0598
社　　址 北京市西城区德外大街 4 号　网　　址 http://www.hep.edu.cn
邮政编码 100120　http://www.hep.com.cn
印　　刷 天津新华二印刷有限公司　网上订购 http://www.landraco.com
开　　本 787mm × 1092mm　1/16　http://www.landraco.com.cn
印　　张 26.5　版　　次 2013 年 11 月第 1 版
字　　数 510 千字　印　　次 2013 年 11 月第 1 次印刷
购书热线 010-58581118　定　　价 38.00 元

物 料 号 38569-00

序

自计算机问世以来，程序设计已取得了长足的进步，人们经历了从机器语言到高级语言的不同阶段；设计了适应不同要求、具备不同风格的程序设计语言；探讨了一些行之有效的设计方法；开发了若干程序测试和验证的手段；与此同时也发展了一些相应的理论，所有这些极大地丰富了这一领域。尽管如此，至今为止程序设计仍属于人类智力活动的范畴。计算机科学界根据不同的实践，对程序设计有着不同的理解，从而产生了不同的流派。以美国斯坦福大学 D. E. Knuth 教授为代表的科学家，认为它是围绕算法设计为核心所展开的智力活动，反映其观点的代表作即为“The Art of Programming”丛书；以荷兰计算机科学家 E. W. Dijkstra 教授为代表的一些科学家，则把它看作一种近似数学的形式推导过程，其代表作为“A Discipline of Programming”；还有一些人仍然把它当作一种纯粹性的智力活动，其代表作为“The Craft of Programming”。众说纷纭，莫衷一是，至今仍未达成共识，给大学计算机课程的教学带来困惑。

进入新的世纪，计算技术的发展日新月异，已经渗入到各个学科，深入到社会的方方面面，然而目前的现状是，一方面各个领域对计算机专业人才的需求不断增加，另一方面和各个专业相关的计算机人才的培养却不尽如人意。一些专业人士认为，既然计算已和各行各业以及个人生活紧密相关，那么计算就是人们必须掌握的技能之一，而不是计算机专业所独有的知识，计算机教育要做的就是探究计算的本质，让所有有需求的人们能够掌握计算的思想和方法，从而在不同领域能够驾驭不同的计算应用。

2006 年，美国卡内基·梅隆大学的 Jeannette M. Wing（周以真）教授首先提出计算思维（computational thinking）的概念，她将计算思维定义为运用计算机科学的基础概念进行问题求解、系统设计以及理解人类行为等的一系列思维活动。她认为源自数学思维和工程思维的计算机思维，与阅读、写作与算术能力一样，也应成为人类的基本技能。她特别强调：计算思维是“人的，不是计算机的思维”，而且，“计算思维是人类求解问题的一条途径，但决非试图使人类像计算机那样去思考”。

计算思维的提出使人们开始反思，开始认识到原先的那种试图让人们像机器那样去思考并形成如同机器那样去解决问题的方式存在着严重的缺陷。为什么我们要“thinking in XXX”，为什么思想要被某种计算机体系结构或某种程序设计语言所限制，为什么不能以

解决问题的自然方式进行思考？为此许多计算机教育工作者开始整理、归纳计算机学科数十年来形成的解决问题的有效方法，在计算机科学教育中进行尝试，以促使以计算思维技能为中心的教学早日成熟。目前计算思维课程的教学以及以计算思维为核心的程序设计等课程的教学日益完善，但如何将计算思维扩展到非计算机专业的课程还有待探索。

近年来人们逐渐认识到作为创新灵魂的科学思维除了理论思维和实验思维外，还应包括计算思维。随着计算机技术的发展，计算思维对各个学科的发展日益重要。事实上，我们已经见证了计算思维对其他学科的重要影响。例如，计算生物学正在改变着生物学家设计新药的方法；计算博弈和计量经济正在决定着经济学家对微观经济和宏观经济的思考方法；关于宇宙的计算模型也在改变着物理学家探索世界的方式。

计算思维概念已经被引入大学第一门程序设计课程的教学。

加州大学伯克利分校对学校原先开设的程序设计课程调查得知，非计算机专业的学生，特别是非理工类学生的反馈意见大多是“Dry. Difficult. Irrelevant. ”（枯燥，难学，专业无关）。经过反思，计算机专业教师认识到按照传统的方法讲解程序设计，学生往往难以接受，容易产生排斥心理。受 UML 创始人 Grady Booch 关于计算机科学的讲授应是“passion, beauty, joy and awe”（激情、美丽、乐趣、敬畏）的启发，伯克利分校于 2009 秋季开设了研讨课“The Beauty and Joy of Computing”（计算的美丽和乐趣），教学目标是使程序设计对于学生来说是“Fun. Easy to learn. Can relate to it.”（有趣，易学，相关）。一学期探索的结果表明，该课程的教学内容和教学方式获得学生的好评。由此，自 2010 年起，伯克利分校决定使用该课程代替原先向全校开设的“符号计算引论”（Introduction to Symbolic Programming）课程，向所有没有编程基础的计算机专业和所有非计算机专业学生开放。课程涉及抽象、递归、仿真、并发、博弈等计算思维的概念，不仅讨论了计算技术如何改变世界，计算和社会的相互作用，Twitter（社交网络和微博服务）的工作方式，以及 CS+X 的学科形态，还涉及对云计算和未来计算等热点问题和前缘发展方向的探讨。伯克利分校通识课程使用的是课程组基于 MIT 的 Scratch 自行开发的 BYOB 语言，BYOB 是一种可视化的积木式语言，也就是通过对构成程序的基本构件的积木式的简单拖放，就可由小积木块构造大的积木，直至整个程序。使用 BYOB 的目的是用户界面友好，易上手，学习有成就感，另外容易形成好的编程风范。课程后期，学生可以选择学习 Python 或 Scheme 语言。该课程近期被美国大学理事会选为 First Course in Computing 的五个试点课程之一。

普渡大学的计算思维引论课程“Introduction to Computational Thinking”是美国自然科学基金委 CPATH（Pathways to Revitalized Undergraduate Computing Education，本科计算机教育复兴之路）项目下课题 SECANT（Science Education in Computational Thinking，基于计算思维的科学教育）的重要组成部分。这是一门由计算机科学、物理学、统计学和化学等专业的教师一起协作开设的课程。按照普渡大学的规定每个学科专业的学生至少需选择一门计算相关课程。课程目标是借助基本程序设计方法、数据和数据管理理论以及仿真和

可视交互技术向学科专业的学生传授计算思维概念。课程内容由基本工具，计算工具、方法和结构以及计算机科学 3 个模块组成，第一个模块主要介绍程序设计语言 Python；第二个模块涉及计算物理、计算化学等内容，包括数据结构中一些基本概念如图、树和网络等，还涉及了计算机仿真应用；第三个模块涉及的是与计算机科学相关的知识，包括面向对象概念、计算复杂度和可计算性的基本原理，还讲述了不同计算模型，包括 DNA 计算和量子计算等科学前缘概念。课程使用 Python 语言，包括 Vpython（3D 建模）、NumPy（数学计算）和 NetworkX（基于图的表示和操作）等 Python 库。教学团队包括计算机专业教师和其他专业的教师，计算机专业方面由 S. E. Hambrusch 教授负责，科学方面由物理专业教授 M. Haugan 与统计学和计算机专业教授 O. Vitek 负责。课程得到美国自然科学基金委 Computer Science Pathways for Educators 项目的资助。

MIT 久负盛名的 6.00 课程“Introduction to Computer Science and Programming”，20 多年来一直使用 Scheme 语言来阐明计算机程序的构造并予以解释，使用教材为《计算机程序的构造和解释》（SICP）（SICP 的作者还著有《经典力学的构造和解释》一书，试图使用 Scheme 语言来描述经典力学的基本原理，可以说是计算思维概念的先行者）。2007 年，受各种因素影响，MIT 对 6.00 课程进行了重大改革，使用学生更加容易理解和掌握的 Python 语言作为工具来讲解程序设计和计算机科学。课程目标是使学生理解计算在问题求解中的作用，明确课程的主旨是帮助所有学生，包括那些不打算进入计算机专业学习的学生，具备能编写求解实用问题的简单程序的能力。改革后的课程核心内容包括基本程序设计技术、递归算法设计和动态规划技术、面向对象的基本概念、计算机仿真和数理统计学原理，课程的最后一课则从计算思维的角度对整个课程进行总结。

从上述计算思维相关课程看，加州大学伯克利分校注重学习的趣味性，通过展示计算之美来讲授计算思维；普渡大学则侧重于计算在其他学科中的应用，通过邀请其他学科的学者共同讲授课程，促进学生计算思维技能的发展；MIT 则通过程序设计的讲授引导学生了解计算机科学，帮助学生获得通过编程解决实际问题的能力，树立自信心。

另外，计算思维还融入了其他学科的课程。在马萨诸塞州的惠顿学院，计算机系教授马克·勒布朗开设了基于计算思维的跨学科课程“Computing for Poets”（http://cs.wheatoncollege.edu/~mleblanc/131/）。该课程和英语系开设的有关 J.R.R.托尔金文学和盎格鲁-撒克逊文学两门课程绑定在一起，在课程中学生学习 Perl 语言，并以基于计算思维的概念来分析相关的文本资料。美国哈佛大学开设的跨学科计算思维课程“Economics and Computation”，融合了经济思维（economic thinking）和计算思维的基本概念，涉及电子商务、社会网络、群体智能和网络系统等。使用的参考书为 David Easley 和 Jon Kleinberg 编写的 Networks, Crowds, and Markets: Reasoning About a Highly Connected World（Cambridge University Press，2010）。目前应用计算思维和使用 Python 讲授计算物理和计算化学的课程有很多，如 Oregon 大学物理系开设的 Computational Scientific Thinking 课程作为计算物理

的先修课程，在他们撰写的教材中系统阐述了计算物理和计算机科学等学科的关系等。

2009年，上海交通大学程序设计思想与方法课程教学小组，针对课程改革的需求，在分析国外相关课程设置情况、广泛听取学生意见的基础上，提出了基于计算思维的程序设计通识课程教学方案，放弃以讲授某种具体程序设计语言及其应用为目标的旧教学大纲，探索课程核心内容为“真正”程序设计“思想”和“方法”的教学思路，具体来讲，就是改革目前的程序设计课程教学形式，以“发现问题—分析问题—寻求问题多种解决方案—对各种解决方案比较选优—实现解决方案”的“问题求解驱动式程序设计”训练方法，尽可能接近解决实际问题的模式。此外，从典型（基本）问题和综合（复杂）问题两个层面，揭示领域相关和领域无关各种问题的本质，并有针对性地讲授不同计算机制解决问题的方法。学校教务处非常重视，经过多次研究讨论，最终同意在新学期试教。三年多来，融计算思维的程序设计思想与方法的课程改革和教材编写工作受到了上海交通大学985三期课程建设、上海市教委重点课程建设、教育部高教司“理工类高校以计算思维为导向的大学计算机系列课程改革及实践”项目的子项目的资助，课程组成员也应邀多次在国内计算机课程教学论坛上介绍经验，最重要的是课程受到了学生的好评，选修课程的学生人数每学年都达到了一千余人。

陆朝俊老师从头开始参与课程的改革和探索，他撰写的这本书是多年来教学的积累和结晶，教材内容吸收了CMU、MIT、Stanford等高校相关课程教材和参考书的精髓，并按照国内教学要求进行了提炼和拓展，教材以计算思维为主线介绍了程序设计的思想与方法，包括模块化编程、图形编程和面向对象编程等，涉及了数据和大（量）数据前沿话题，“计算+X”一章还针对不同的学科讨论了其与计算的融合。相关内容已在上海交大讲授三年并按照学生反馈进行了精心的修改。

本教材可以作为高等院校计算机专业和非计算机专业“计算思维导论”、基于计算思维的“程序设计思想与方法”或“计算机科学引论”等课程的教材。

孙永强　黄林鹏

2013年8月1日

前 言

2006 年 3 月，美国计算机科学家 Jeannette M. Wing（周以真）在 CACM 上发表文章《计算思维》（Computational Thinking），主张计算机科学家应该向大学新生讲授一门关于如何“像计算机科学家那样思考”的课程，这门课并非仅为计算机科学专业学生开设，更重要的是面向所有非计算机专业的学生，甚至是面向中小学学生。进行计算思维教学的目标是使计算思维像阅读、写字、算术一样成为每个人的基本技能。

所谓“计算思维”，是指运用计算机科学的基础概念、思想和方法去解决问题时的思维活动，涉及如何在计算机中表示问题，如何让计算机通过执行有效的算法过程来解决问题。计算机原本只是人们解决问题的工具，但当这种工具在几乎每一个领域中都得到广泛应用后，工具就会反过来影响人们的思维方式。因此，将计算思维向所有人进行普及，使普通人群也能像计算机科学家那样利用计算机来解决自己生活、工作中的问题，对于人们适应未来的、计算机无处不在的社会，具有重要意义。

上海交通大学为全校学生开设一门称为“程序设计思想与方法”的通识课程已有多年，从 2010 学年秋季学期开始，我们对该课程进行改革，试图将它转变成计算思维课程。由于计算思维是一门崭新的课程，国内外都没有合适的教材，甚至计算思维课程应当讲授的内容也没有定论，这促使我们按照自己的理解编写了这本计算思维教材。

目标

本书向计算机专业和非计算机专业的学生介绍计算机科学的基本概念、思想和方法，目的是使学生理解计算机科学家的思维特点和方式，并最终能够利用计算机解决自己专业领域的问题。

内容

本书内容覆盖利用计算机解决问题的全过程。

第 1 章首先界定“计算”的含义，然后介绍“计算思维”的基本内容。计算是指利用计算机解决问题的过程，而非传统意义的数学计算，其实质是“算法化”，即按照一定的步骤执行基本指令的过程。为了让学生实践所学到的计算机问题求解的思想和方法，需要利用某种编程语言来实现算法，本书采用 Python 语言作为编程的教学工具。

计算机可以看作是信息处理机器，所有问题的解决都是对特定信息进行特定处理的过程。第 2 章和第 6 章介绍如何在计算机中表示现实世界信息，其中第 2 章介绍简单信息的表示，包括数值、字符串等；第 6 章介绍复杂信息的表示，包括各种集合体数据和数据结构。

第 3 章介绍如何表示对信息的处理过程，包括顺序、条件、循环等控制流程的表示以及结构化编程的思想。第 4 章介绍如何将信息处理过程按照良好的结构组织起来，模块化编程和自顶向下设计可以帮助我们建立复杂问题的处理过程。

第 5 章介绍图形编程。图形是传达信息的最高效的手段，在利用计算机解决问题时经常用图形来实现可视化计算，因此图形编程的重要性不言而喻。同时，早早地让学生学会图形编程并编制一些有意思的程序，能够激发他们的学习兴趣。第 8 章介绍的图形用户界面是图形编程的进一步延伸。

第 7 章详细介绍当前流行的面向对象编程。面向对象不只是一种编程范型，它还是一种强大的思维工具，可以说是本书的重点内容。

传统计算都是确定性的，第 9 章介绍两个与不确定性相关的内容。计算机模拟是在各行各业中广泛应用的方法，本章介绍如何利用蒙特卡罗方法模拟现实世界中的不确定性。另外，本章还简单介绍了多线程并发计算。

第 10 章介绍算法设计和分析。这章内容涉及理论计算机科学，旨在使读者了解计算机的计算能力和局限。

第 11 章介绍所谓“计算+X”，说明计算机与各专业的结合能够形成多种交叉学科，同时也证明了计算思维课程的重要意义。作者不可能了解各专业的知识，所以本章只能是浅尝辄止。

为什么选用 Python

由于计算思维课程要面向广大的非计算机专业学生，我们希望他们能够轻松地学会一种编程语言，以便动手实践随后学到的知识。Python 语言非常简单，易学易用，可以让学生在第一堂课就学会编写简单程序。另外，我们希望能用直观、形象的方式来展开课程教学，Python 语言正好能满足我们的需求。Python 是一种编译/解释混合的高级编程语言，这使我们在课堂上可以通过会话方式与 Python 解释器进行交互，即时演示教学内容。最后，Python 语言支持我们希望在本课程中介绍的各种特性，如结构化编程、面向对象编程、图形和 GUI 编程多线程等。

事实上，Python 语言是当今最受欢迎的程序设计语言之一。由于 Python 语言的简洁、易读以及可扩展性，国外许多知名大学研究机构都用 Python 教授程序设计课程和进行科学计算。Python 语言及其众多的扩展库所构成的开发环境十分适合处理实验数据、制作图表

以及开发各种应用程序。

要说明的是，尽管本书介绍了很多 Python 语言的知识，但本书并不是“Python 语言”教材，没有像编程语言教材那样介绍 Python。更多关于 Python 语言的内容，请参考专门的资料。

教学建议

首先，本教材包含的内容适合各专业学生的学习。对于非计算机专业的学生，可以忽略那些较为深入的、涉及更多技术细节的内容，本书为这样的内容加上了“*”标记。

其次，在课堂上演示所教内容对于非计算机专业学生来说具有良好的效果，本书在编写时充分考虑了这一点。在书中，有许多以下列形式出现的代码：

```
>>> print "Hello, World!"
Hello, World!
```

其中特意保留了 Python 解释器提示符“>>>”（并不是自己输入的），以提醒教师这样的代码可以当场演示。必要时用黑体表示输入的内容，以区别于 Python 显示的内容。任何阅读本书的读者都可以模仿这样的代码，边读书边动手实践。

第三，教师讲授时不必严格按照本书的章节次序，可以根据需要提前或延后某些内容。例如，第 2 章中的布尔类型可以与第 3 章的条件语句结合起来讲解。又如，第 3 章的异常处理可以延后介绍，以便尽早完成基本的条件和循环语句的讲解，让学生能尽快编写有用的程序。再如，第 9 章介绍的模拟方法可以适当提前介绍，因为学会这个方法能使学生解决许多有趣的问题。

致谢

上海交通大学的“程序设计思想与方法”课程改革得到了上海交通大学 985 三期课程建设、上海市教委重点课程建设、教育部高教司“理工类高校以计算思维为导向的大学计算机系列课程改革及实践”项目的子项目“基于计算思维的程序设计方法课程改革”的资助，在此表示感谢。

上海交通大学计算机系有许多教师从事“程序设计思想与方法”课程的教学，作者在与他们的讨论、交流中获益匪浅。尤其是本课程改革的负责人黄林鹏教授，向作者提供了很多资料、建议和外校同行们的做法，并抽空审阅本书书稿，指出了若干错误之处，提出了一些修改建议。衷心感谢所有同事的支持和帮助。

感谢来自各专业的学生，他们在课堂内外的表现和提问，使作者获得了向非计算机专业学生讲授计算思维课程的经验。而很多学生在期末大作业中利用所学知识解决自己专业问题，也令作者很欣慰，说明本课程确实达到了目的。

感谢导师孙永强教授、师兄黄林鹏教授为本书作序，序言介绍了国际上计算思维课程

的发展脉络以及上海交通大学课程改革的思路，建议读者一读。

最后要感谢妻子和女儿的支持，忙碌的写作使作者有些忽略了对她们的照顾。

由于作者水平有限，书中错漏之处一定不少，恳请读者不吝赐教！欢迎读者将反馈意见发到作者的电子邮箱：lu-cj@cs.sjtu.edu.cn。

陆朝俊

2013 年 5 月

目 录

第 1 章　计算与计算思维

计算是利用计算机解决问题的过程，计算机科学是关于计算的学问。计算机科学家在用计算机解决问题时形成了特有的思维方式和解决方法，即计算思维。本章介绍计算的基本概念和计算思维的基本内容，而本书的其余章节将围绕计算与计算思维这个中心展开详细讨论。

1.1　什么是计算

1.1.1　计算机与计算

计算机是当代最伟大的发明之一。自从人类制造出第一台电子数字计算机，迄今已近 70 年。经过多年的发展，现在计算机已经应用到社会、生活的几乎每一个方面。人们用计算机上网冲浪、写文章、玩游戏或听歌、看电影，企业用计算机管理组织机构及业务流程、设计制造产品或从事电子商务，大量机器被计算机控制，手机与计算机之间的界限越来越模糊，……总之计算机似乎无处不在、无所不能。那么，计算机究竟是如何做到这一切的呢？为了回答这个问题，需要了解计算机的工作原理。

提到计算机，人们头脑中会首先浮现出显示器、键盘、主机箱等一堆设备——计算机硬件。了解一些硬件设备的基本知识自然是需要的，不过从学习用计算机解决问题这个层面看，并不需要掌握太多底层硬件知识。在此仅介绍现代计算机的主要功能部件，目的是让读者了解用计算机解决问题的计算机制。现代计算机的主要功能部件如图 1.1 所示。

CPU、指令与程序

中央处理器（CPU）是计算机的计算部件，能够执行机器指令，或简称指令

(instruction)。每条指令表达的是对特定数据执行特定操作。某种 CPU 能执行的全体指令是由该 CPU 的制造商设计并保持固定不变的，称为该 CPU 的指令集。例如，Intel 公司为它的 80x86 系列处理器设计了上百条的指令。

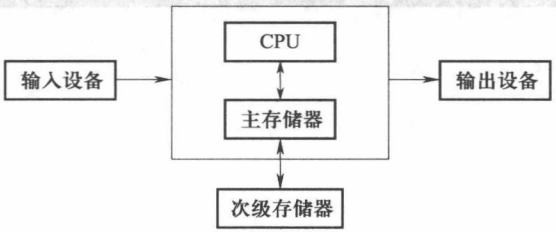

图 1.1　计算机的主要功能部件

外行人也许会以为，计算机功能如此强大，必定是因为它能执行功能强大的指令。然而事实并非如此。即使是当今技术最先进、计算能力最强大的计算机，它的 CPU 也只会执行一些非常简单的指令，例如将两个数相加、测试两个数是否相等、把数据放入指定的存储单元等。

由于每条机器指令都只能完成很简单的操作，因此仅靠少数几条指令是无法实现复杂操作的。但是，如果将成千上万条简单指令组合起来，却能解决非常复杂的问题！亦即，复杂操作可以通过执行按特定次序排列的许多简单操作而实现。这种由许多指令按次序排列而成并交给计算机逐条执行的指令序列称为*程序*(program)。为了用计算机解决问题，把问题的解法表达成一个指令序列(即程序)的过程，称为*程序设计*或*编程*(programming)。可见，计算机所做的所有神奇的事情，都是靠一步一步执行的、平凡而乏味的简单指令序列实现的。计算机一点也不神奇，它唯一会做的事情就是机械地执行预定的指令序列。

存储器

存储器是计算机的记忆部件，用于存储数据和程序。

存储器分为主存储器和次级存储器，它们是用不同的物理材料制成的。CPU 只能直接访问主存储器，也只有主存储器才能提供与 CPU 相匹配的存取速度。但主存储器需要靠持续供电来维持存储内容，一旦断电，存储的数据或程序就会消失。为了持久保持存储信息，可以使用即使断电也能维持存储的次级存储器，如当前普遍使用的磁盘。CPU 不能直接访问次级存储器，次级存储器上的数据或程序必须先送到主存储器中，才能被 CPU 存取或执行。次级存储器的读写速度远远低于主存储器，这个差别极大地影响了用计算机解决问题时所使用的方法。

现代计算机在体系结构上的特点是：数据和程序都存储在主存储器中，CPU