教育部大学计算机课程改革规划教材

C语言程序设计

（第二版）

C YUYAN CHENGXUSHEJI

甘勇　李晔　卢冰◎编著

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

教育部大学计算机课程改革规划教材

C语言程序设计（第二版）

甘 勇 李 晔 卢 冰 编著

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书以程序设计过程为主线，以问题和案例引入内容，围绕问题的解决来讲解C语言及程序设计。全书共分13章，主要内容包括：引言、简单C程序设计、分支结构、循环结构、函数、数组、字符数组与字符串、指针、结构、指针进阶、C预处理、文件及计算思维与常用算法。

本书内容全面，知识点详尽，适合作为高等学校各专业C语言程序设计课程的教材，也可作为从事计算机相关工作的人员的参考书。

图书在版编目（CIP）数据

C语言程序设计/甘勇，李晔，卢冰编著. —2版. —北京：中国铁道出版社，2015.9

教育部大学计算机课程改革规划教材

ISBN 978-7-113-20707-6

Ⅰ. ①C… Ⅱ. ①甘… ②李… ③卢… Ⅲ. ①C语言－程序设计－高等学校－教材 Ⅳ. ①TP312

中国版本图书馆CIP数据核字(2015)第195860号

书　　名：C 语言程序设计（第二版）

作　　者：甘勇　李晔　卢冰　编著

策　　划：周　欣　祝和谊　　**读者热线**：400-668-0820

责任编辑：周　欣

编辑助理：祝和谊

封面设计：一克米工作室

责任校对：汤淑梅

责任印制：李　佳

出版发行：中国铁道出版社（100054，北京市西城区右安门西街 8 号）

网　　址：http://www.51eds.com

印　　刷：北京市昌平百善印刷厂

版　　次：2014 年 9 月第 1 版　　2015 年 9 月第 2 版　　2015 年 9 月第 1 次印刷

开　　本：787 mm×1 092 mm　1/16　印张：22.75　字数：534 千

书　　号：ISBN 978-7-113-20707-6

定　　价：43.00 元

序一
PREFACE

信息网络正以铺天盖地之势步入学校，走进家庭，风靡天下：男女老少用手机收发微信、购物、咨询、问路、娱乐等，成为街头巷尾的一道风景线，数以亿万计的手机悄然成为中国网民的“首选上网终端”，人们在不知不觉中享受到网络带来的快乐、方便与效率，同时也感受到异乎寻常的现代文化的魅力。学习计算机，早已超出技术的层面，上升到现代社会文化的层面。讲文化要以科学为基础，学科学要站到文化的高度。有关信息的收集、处理、分发、转换、存储、挖掘、评价等的知识和技术，愈发显现其重要性。信息技术和网络的发展如日中天，超乎寻常，对于新东西，谁敏感，谁学得快，谁就主动得多。形势要求大学计算机课要和大学数学、大学物理一样，作为基础课来学。

2012 年教育部高教司正式发文（教高司函〔2012〕188 号）批准立项 22 个“大学计算机课程改革”项目，启动了新一轮大学计算机基础教育改革。“以计算思维能力培养为主线的理工类专业大学计算机课程改革研究”是其中一个项目，项目组组织计算机教育专家和具有丰富教学经验的一线教师，对计算机基础教育中的教学内容、教学方法，以及新一轮计算机基础教育改革的教材如何编写等问题，展开了深入研究与探索，特别着力研究了能力定义、能力模型、专业能力、思维能力和行动能力等基本概念，研究了以能力培养为重点的计算机应用能力的体系框架。通过调查研究、总结经验和相互交流，结合中国大学自己的实际，找到如下五点共识，作为设计课程和编写新一轮教材的思想和行动基础：

（1）提升大学计算机基础教育重要性的认识，学习和掌握信息科学与技术，在高水准人才的知识结构中占有重要的地位。

（2）计算机高速发展的不竭动力是应用。对于学生，说到底，学是为了用。因此，大学计算机基础教育一定要实施面向应用的教育，上什么和教什么一定要以学生的专业需求为导向。

（3）学习计算机需要了解这一学科的内在规律和特征，“构造性”和“能行性”是计算机学科的两个最根本特征。与构造性相应的构造思维，又称计算思维，指的是通过算法的“构造”和实现来解决一个给定问题的一种“能行”的思维方式，或者说计算思维其实就是解决问题时的计算模拟方法论。

做任何事情都须讲求方法，让计算机帮我们做事，更要讲求方法。科学方法基于科学思维：理论思维（逻辑思维）、实验思维（实证思维）、计算思维（构造思维）。计算思维这个词，是 20 世纪 50 年代之后，计算机发挥了人类期望的强大计算功能之后，被人们认识、研究和提炼出来的。计算思维目前有着诸多的定义或描述，究其本质是“抽象”和“自动化”。数学抽象是针对现实世界的量的关系和空间形式来进行的。计算思维中的抽

象与传统数学相比更为复杂和实用，抽象的好坏和是否能够实用，要看计算机能否快速地自动化地完成人们预想的计算任务。当今学界的有识之士提出：既然计算机已经成为“人类通用智力工具”，那么计算思维对每个学生都有普适意义。在实施面向应用的计算机教育时，将计算机解决问题的思路归纳出来，点化学生，使其感悟计算思维要素之妙，就能起到“举一反三”的作用。

（4）工具是重要的，人之所以能进化为现代人，不断发明和成功创造工具起了绝对重要的作用。计算机是“人类通用智力工具”，尽管它问世的时间还不长，但是它所展现出来的智能化的趋势已令世人叹为观止。对于这样一个工具，我们应该怎么教？谁也不敢说胸有成竹，还需要在教学改革的进程中下大功夫加以研究。

（5）既然是面向应用的课，更应强化理论联系实际的优良学风，引导学生动手动脑，使思维能力和行动能力同步提升，以培养高素质，能解决实际问题的应用型人才。强化理论指导下的实践，是课程取得成功的必由之路。

在专家和一线教师的参与下，新一轮教材编写出来了。我们有理由相信：这会对我国计算机基础教育水平的提高起到推动作用，因为“大学计算机课程改革”项目的研究与实施，让大家有了明确的目标和正确的思路，具备了编写新教材的良好的思想和行动基础。

吴文虎

2014 年 8 月 12 日

序二
PREFACE

大学计算机基础教育已经发展了三十年，已成为我国高等教育教学的重要组成部分。它以培养学生应用计算机技术解决实际问题的能力为目标，使之成为在各自专业领域熟练掌握计算机应用能力的专门人才。大学计算机基础教育对实现我国信息化战略目标和提升全国人民信息素养起着举足轻重的作用。伴随着新一轮大学计算机基础教育教学改革，大学计算机基础教育正在进入一个新的阶段，迫切需要有新的突破。在这样的背景下，由教育部高教司和教指委共同提出和推动，以计算思维为切入点的新一轮大学计算机基础教育教学改革在2012年正式启动。

“以计算思维能力培养为主线的理工类专业大学计算机课程改革研究”是2012年教育部高教司正式发文（教高司函〔2012〕188号）批准立项的“大学计算机课程改革”项目之一，自批准立项以来，项目组在项目负责人的带领下，在众多高校相关教师的参与下，在中国铁道出版社的大力支持下，开展了卓有成效的研究工作。

项目组由国内著名计算机基础教育专家带领并指导，由从事大学计算机基础教育教师、教育学科学者与行业企业专家合力协同研究，一批年轻的博士、博士后教师牵头子课题成为项目研究的骨干，从理论到实践对大学计算机基础教育进行深入的研究，提出了新的大学计算机基础教育教学改革指导思想和课程开发方法，并开发了若干典型课程和教材。

本项目设计定位为：面向非研究型本科院校理工科各专业的大学计算机基础教育，重点包括：

（1）传统大学分类中的教学型、教学研究型大学（学院）等；

（2）近年在本科大学中发展起来的应用型大学（学院）；

（3）现代技术与职业教育体系中的应用技术大学。

以上几类大学（学院）在我国1 200多所本科大学（学院）中约占千所左右，并在本项目研究中统称为非研究型本科院校。

本项目研究以教育部高等学校计算机基础课程教学指导委员会推动新一轮大学计算机教育改革，提高教学质量为总目标；以理论与实证研究为基础，以实践层面的教学成果和资源转化为目标，注重学生信息素养、计算思维和计算机应用能力的提升，认真规划大学计算机课程的知识结构、课程体系和教学方案，完成课程设计和教材开发；建设相应的教学资源和学习平台。

在理论研究层面提出了计算思维的全面定义，研究了计算思维的培养方式和目的；提出了大学计算机基础教育能力模型，明确了知识、能力、素质之间的关系；提出了大学计算机课程设计方法和新的大学计算机基础课程体系设计慨念；设计了大学生计算机基本应用能力标准。在实证研究层面，进行了大学生计算机应用能力状态与企业应用需求的实证

研究和大学生计算机应用能力状态的研究。在理论与实证研究基础上，明确了大学计算机基础教育必须传承的四项原则，提出新一轮大学计算机基础教育教学改革的五个目标。在以上研究基础上开发了若干门典型课程和教材。

项目组的教师经过多次研讨、交流，在对新一轮大学计算机课程改革的理论与实证研究基础上，按照项目组提出的课程体系框架，从顶层设计着手，进行课程开发和教材编写。依据新一轮大学计算机课程改革的理念，改革的大学计算机课程大体可分为三类：

第一类为基于计算机应用的课程。其特征是以计算机基本技能或技术为基础，反映新技术发展和应用需求，并在此基础上以解决问题为目标，通过项目或案例培养思维和行动能力，着重注意融入计算思维。这类课程和教材主要有“大学计算机基础”“大学计算机”“C语言程序设计”“VB程序设计”“计算机网络技术”“软硬件综合应用技术”等。

第二类为计算机学科融入其他学科内容的学科融合性课程。近年伴随信息技术的发展出现了很多以计算机为基础学科，与其他学科交叉共融的理论、方法和应用领域，并在实际中广泛应用，如“大数据技术”，其应用已渗透到经济社会、生产生活各个领域，大数据已成为重要的战略资源，而大数据技术的本质是计算机技术与数学方法的融合。一些发达国家的高等教育中已经对各专业学生开出了与此相关的课程，本项目开发的“数据科学与大数据技术”课程和教材将成为新的大学计算机基础课程体系组成部分，也可以代替原来的“大学计算机”或“数据库技术与应用”等课程。“多媒体艺术设计”是另一门本项目研发的艺术与计算机技术相结合的大学计算机基础课程。

第三类为计算机学科与专业结合的课程。这类课程是计算机技术直接面向专业应用的课程，所谓“专业应用”是指可以面向学科专业应用，如最近已见报道的“计算医学”，就是面向医学专业学科的应用；也可面向专业指向的行业、产业应用，如“计算农业”就是面向农业产业的应用；还可以面向某些领域的应用，如本项目组正在着手研发的“计算工程”就是面向工程领域的应用。这类课程开拓了计算机基础与专业应用相互融通新的大学计算机基础课程方向。

以上课程都应在教学中注意提升学生解决问题的通用能力，并在其中训练学生的思维和行动能力，作为大学计算机基础课程，当然也应将计算思维能力培养作为自己重要课程目标。

新一轮大学计算机基础教育改革的重要意义毋庸置疑，不仅所有相关教师要更新观念、提升能力、积极参与，院校的各级领导和管理部门、省市与教育部各级相关管理机构与领导更应给予关注。加强研究，勇于探索，营造氛围，制定政策，为新一轮大学计算机教育改革保驾护航。祝愿改革中的大学计算机基础教育能在飞速发展的信息时代发挥更重要的作用。

高林

2014年8月

第二版前言
FOREWORD

C语言程序设计是很多理工科专业，尤其是计算机专业学生必修的一门专业基础课。C语言从产生到现在，已经成为最重要和最流行的编程语言之一。学习、掌握C语言成为了每个计算机技术人员的一项基本功，也是在计算机领域中进一步学习和工作的基础。

本书在第一版的基础上，汲取了教学实践中众多师生的反馈意见和建议，调整了部分知识点的讲解顺序，改进了部分内容的叙述方式，并对部分同容进行了增删，修改了部分例题的解题方法。

程序设计课程的核心是计算机解题的思维方式，而思维要借助工具（如C语言）来表达。C语言本身功能强大、内容复杂，以语法为中心的C语言书籍作为工具书是可行的，但作为程序设计课程的教材是有缺憾的，“重语法轻思维”是舍本逐末的做法。本书最大的特点是向“纵深”结构发展，旨在“程序设计”，重在培养学生在编程中解决问题的思维能力和编码能力，并没有试图覆盖C语言的所有语法知识点，语法知识点的引入是为解决问题服务的。带*号的章节为自学章节，也是为了提升学生编程能力而设计的，老师可以不讲，但学生必须要看。

本书的亮点在于示例程序的选取，通过问题分析和代码实现向读者传递枚举、模拟、递推、递归、空间换时间等计算思维方式，引导读者理解并践行这些思想，正确高效地使用C语言编程。例如，在循环结构一章，通过阶乘计算、数列求和、斐波那契数列等例子让读者深入掌握“递推”这一计算思维方法，从而充分利用中间结果，降低算法复杂度；通过百钱买百鸡等例子让读者接触“枚举”这一计算思维方法；通过模拟投点法计算圆周率近似值等例子让读者接触“模拟”“随机化”等计算思维方法。

本书由甘勇、李晔、卢冰编著，参编人员有王捷和苏虹。本书在编写过程中得到了贾志娟、赵少林、金保华、朱付保、段赵磊和李灿林等很多老师以及郑州轻工业学院教务处的帮助和支持，作者深表感谢。最后还要感谢所有第一版的读者，感谢你们对这本教材的厚爱及提出的宝贵意见。

由于作者水平有限，书中难免存在疏漏之处，敬请读者批评指正。

编　者

2015年6月

第一版前言
FOREWORD

C 语言程序设计是很多理工科专业，尤其是计算机专业学生必修的一门专业基础课。C 语言从产生到现在，已经成为最重要和最流行的编程语言之一。现在，学习、掌握 C 语言成为了每个计算机技术人员的一项基本功，也是在计算机领域中进一步学习和工作的基础。

在 C 语言程序设计的教学当中，我们参阅过多部国内外的相关教材、讲义，就学生的编程能力以及参加各级竞赛的表现来看，教学方法改革需要与时俱进，因此，有一本适合学生使用的教材非常重要。目前，C 语言教学普遍存在的问题是，很多老师和 C 语言教科书都是围绕知识点进行讲解，过分地强调知识细节，学生也把学习重点和注意力放在了记忆大量的语法细节方面，学习结束后却不会编程。C 语言是为解决问题诞生的，是一种出色的解决问题的工具，掌握一个工具的目的是通过使用它能解决问题。

本书以编程能力的提高为主旨来讲解 C 语言，语法知识点都是伴随着有价值的应用而出现的，强调语法的学习是为了提高用 C 语言解决问题、表达算法思想的能力，书中的例子分为两种，一种为“试试看”类型的例子，比较简单，是为验证或解释某个语法知识而设计的程序；另一种为“解决问题”类型的例子，每个例子都有针对问题的分析和讨论，以帮助初学者了解程序设计过程的实质，理解从问题到程序的思考过程，从而举一反三，快速提高编程能力。带 * 号的章节为自学章节，也是为了提升学生编程能力而设计的，老师可以不讲，但学生必须要看。

C 语言程序设计是一门实践性很强的课程，“纸上谈兵”式的光学不练是学不好 C 语言的。学习程序设计需要大量的编程练习，只有不断地编写程序并改正错误才能真正学会编程。错误与程序设计如影随形，密不可分，只要编写程序，就无法避免错误。之所以在前言中如此强调错误与程序设计的关系，是因为看到太多的同学在学习 C 语言的过程中因为编程中的一个个错误而止步不前、半途而废。事实上每个程序员的成长历程都是一部不断认识错误并纠正错误的纠错史。

本书的编写，也是一个基于实践的教学改革，在使用过程中，我们会继续总结经验、发现问题、不断学习，进一步完善和提高教学和教材水平。

本书由甘勇任主编，李晔、卢冰任副主编，参编人员有王捷、苏虹、段赵磊、贺蕾、张娟娟和贾志娟。本书在编写过程中得到了金保华、尚展垒、钱慎一、朱付保等很多老师，以及郑州轻工业学院教务处、河南农业大学教务处和郑州师范学院教务处的帮助和支持，在此深表感谢。

由于编者水平有限，书中难免存在疏漏之处，敬请读者批评指正。

编　者

2014 年 6 月

目录 CONTENTS

第 4 章 循环结构 / 65

第 5 章 函数 / 109

第 6 章 数组 / 131

第 7 章 字符数组与字符串 / 163

第 8 章 指针 / 187

第 9 章 结构 / 219

第1章 引 言

- 计算机与程序设计语言
- C语言的起源与发展
- C语言的特点
- 一个简单的C语言程序

“C 语言”在程序设计语言中是个闪亮的名字，每个学习计算机及相关专业的人在学习程序设计语言时都会首先关注它。学习 C 语言，具有挑战性却有着高回报，这在我们学习和使用过程中会有深刻体会。C 语言是 20 世纪 70 年代初在贝尔实验室开发出来的一种广为使用的计算机程序设计语言，它从诞生起就主宰整个软件行业，从来没有遇到过挑战，也从来没有离开过，甚至被称为编程语言的万王之王。但是，C 语言究竟有哪些特别之处？为什么国内外大部分高校都把 C 语言作为程序设计入门语言？让我们在对计算机与程序设计语言的诞生、C 语言的起源、特点及其几十年来的发展的介绍中寻找答案吧。

1.1 计算机与程序设计语言

计算机是能够进行计算和逻辑判断的电子设备，2014 年 6 月，TOP500 组织公布了最新全球超级计算机 500 强排行榜榜单，中国国防科技大学研制的“天河二号”超级计算机，再次位居榜首，其峰值计算速度为每秒 5.49 亿亿次，持续计算速度每秒 3.39 亿亿次双精度浮点运算，相当于地球上每个人同时执行几百万次运算。

计算机在被称为计算机程序的机器指令序列的控制下对数据进行处理。而计算机程序中的指令序列是由被称为计算机程序员的人事先制定好的。本书将带领读者探索如何编写计算机程序，命令计算机实现特定功能。

1.1.1 程序存储思想

“程序”一词来自于生活，通常指完成某件事务的一种既定方式和过程。在日常生活中，可以将程序看成是一系列动作执行过程的描述。而计算机程序是人们为了让计算机解决某个问题而编写的一系列有序指令的集合。

冯·诺依曼最先提出了在数字计算机内部的存储器中存放程序的概念，冯·诺依曼理论的要点是：数字计算机的数制采用二进制，计算机应该按照程序顺序执行，这是所有现代电子计算机的理论基础，存储程序控制原理又称冯·诺依曼原理。由于他对现代计算机技术的突出贡献，冯·诺依曼又被称为“计算机之父”。

计算机之所以采用二进制编码，是因为二进制只有 0 和 1 两个数码。可以表示 0、1 两种状态的电子器件很多，如开关的接通和断开、晶体管的导通和截止、电位电平的高与低等都可表示 0、1 两个数码。早期的程序员们将 0、1 数字编程的程序代码打在纸带或卡片上，1 打孔，0 不打孔，再将程序通过纸带机或卡片机输入计算机，进行运算。

存储程序思想——把计算过程描述为由许多指令按一定顺序组成的程序，然后把程序和数据一起输入计算机，计算机便可自动地从一条指令转到执行另一条指令，对已存入的程序和数据进行处理后，输出结果。

为实现存储程序思想，计算机必须具备五大基本组成部件，包括：

- 输入数据和程序的输入设备；
- 记忆数据和程序的存储器；
- 完成数据加工处理的运算器；
- 控制程序执行的控制器；
- 输出处理结果的输出设备。

这是现代计算机的模板，被称为“冯·诺依曼结构”。

1.1.2 程序设计语言的发展

虽然计算机技术发展很快，但存储程序思想至今仍然是计算机的基本工作原理。这一原理决定了人们使用计算机的主要方式——编写程序和运行程序。科学家们一直致力于提高程序设计的自动化水平，改进用户的操作界面，提供各种开发工具、环境与平台，都是为了让人们更加方便地控制计算机，可以少编程甚至不编程来使用计算机。

1. 机器语言

计算机能够直接读懂的语言是机器语言，机器语言是直接用二进制代码指令表达的计算机语言。机器语言的指令是用 0 和 1 组成的一串代码，它们有一定的位数，并分成若干段，各段的编码表示不同的含义。例如，某台计算机字长为 16 位，即由 16 个二进制数码组成一条指令或其他信息。16 个 0 和 1 可组成各种排列组合，通过线路变成电信号，让计算机执行各种不同的操作。例如，某种计算机的指令为 1011011000000000，它表示让计算机进行一次加法操作；而指令 1011010100000000 则表示进行一次减法操作。

计算机可以直接识别机器语言，不需要进行任何翻译。每台机器的指令，其格式和代码所代表的含义都是硬性规定的，机器语言对不同型号的计算机来说一般是不同的。由于机器码是用许多二进制数表示的，用机器语言编程必然很烦琐、难记忆、易出错，非常消耗精力和时间，并且难以检查程序和调试程序，工作效率低。

2. 汇编语言

为了使程序员摆脱机器语言的束缚，提高编程效率，计算机科学家进行了一种改进：用一些简洁的英文字母、符号串来替代一个特定指令的二进制串，例如，用 ADD 代表加法，SUB 代表减法等。这样一来，程序变得易于理解，纠错及维护也变得比较方便，这种程序设计语言被称为汇编语言。

汇编语言大部分语句直接对应机器指令，执行速度快、效率高、代码体积小，主要用在存储器容量有限但需要快速和实时响应的场合，如仪器仪表和工业控制设备中。

虽然汇编语言较机器语言已有很大的改进，但仍是面向机器的语言，主要缺点是：涉及太多机器资源的细节、依赖于机器硬件、移植性不好。

3. 高级语言

由于汇编语言依赖于硬件体系，且助记符量大、难记，于是人们又发明了更加易用的高级语言。这种语言接近于数学语言或人的自然语言，同时又不依赖于计算机硬件，编出的程序能在所有机器上通用。经过努力，1954 年，第一个完全脱离机器硬件的高级语言——FORTRAN 问世了，这是程序设计语言发展史上的一个分水岭，人们把机器语言和汇编语言称为低级语言（它与计算机硬件的距离比较近，但不便于人理解、不便于编写程序），把以后发展起来的语言称为高级语言。

高级语言的编写方式更接近人们的思维习惯，例如，可以用“+”来表示加法，用“-”表示减法，并且它编写的程序具有一定的通用性。低级语言涉及计算机硬件细节，所以不具有通用性。高级语言远离机器语言，与具体的计算机硬件关系不大，因而写出来的程序可移

植性好、重用率高。要想在某一台计算机上运行用高级语言所编写的程序，该计算机只需要提供该语言的翻译系统即可。

但是，一般的高级语言难以实现汇编语言的一些功能（汇编语言可以直接对硬件进行操作），人们需要有一种既有高级语言的易编写、可移植性好等特性，又具有汇编语言精炼和接近硬件的特性，在这种情况下C语言应运而生。

1.2 C语言的发展简史

C语言，从诞生之初就在程序员中备受青睐。目前，C语言编译器普遍存在于各种不同的操作系统中，例如UNIX、Microsoft Windows及Linux等。C语言的设计影响了许多后来的编程语言，如C++、Java、C#等。

1.2.1 C语言的起源

20世纪，人们一直在为贝尔实验室的发明欢呼，那里是7项诺贝尔奖的诞生地。Ken Thompson和Dennis M. Ritchie（见图1-1）从大学毕业后就进入到贝尔实验室工作直到退休，传奇的C语言和UNIX操作系统也由此诞生。

图1-1 Ken Thompson和Dennis M. Ritchie

1964年，Thompson参与了贝尔实验室、麻省理工学院以及通用电气公司联合开发的一套多用户的分时操作系统，名叫Multics。在开发Multics期间，Thompson创造了名为Bon的程序设计语言（简称B语言）。Thompson身为优秀的设计师，同时又是一名游戏爱好者，他设计了一款电子游戏——Space Travel，该游戏可运行于Multics操作系统之上。1969年，贝尔实验室撤出了Multics计划。Thompson决定独自在一台被丢弃的PDP-7上写一个挤干了泡沫的Multics操作系统，并在此操作系统下重写了他的Space Travel游戏。这一操作系统被同事戏称为Uniplexed Information and Computing System（UNICS），后来改称为UNIX。

UNIX的出现开始并不为大家所看好，但是却引起了贝尔实验室另一位同事的注意，这就是Dennis M. Ritchie，Dennis主动加入进来共同完善这个系统。至此一场轰轰烈烈的UNIX的传奇时代才真正拉开了序幕。1972年，他们联手将UNIX移植到当时最先进的大型机PDP-2上，由于UNIX非常简洁、稳定与高效，以至于当时大家都放弃了PDP-2上自带的DEC操作系统而完全改用UNIX，这时的UNIX已经开始走向成熟。随着UNIX的需求量日益增加，Ken与Dennis决定将UNIX进一步改写，以便可以移植到各种不同的硬件系统。由于UNIX的源代码中不少是用