

#### 规格严格 功夫到家



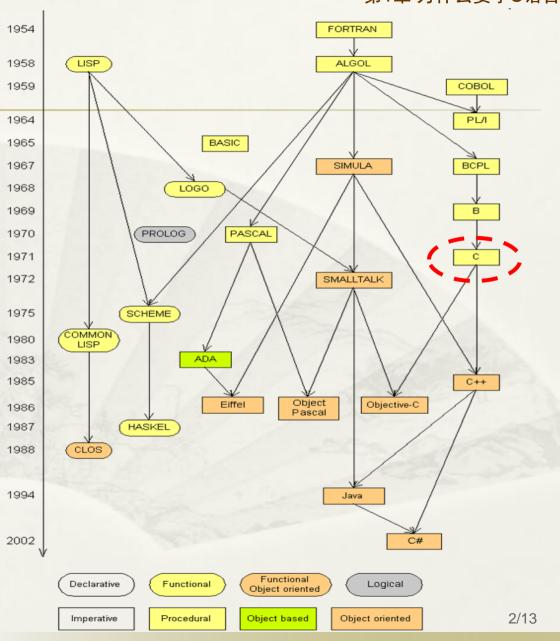
## 第1章 为什么要学C语言

哈尔滨工业大学 计算机科学与技术学院 叶 麟

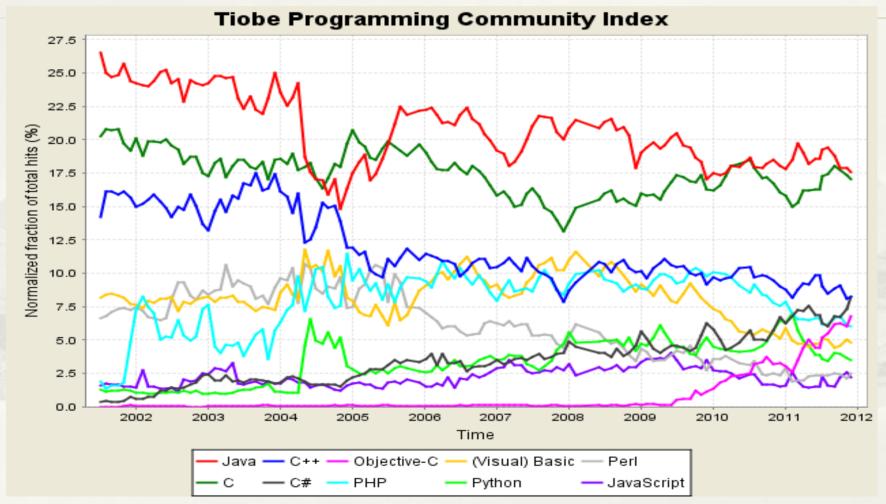
#### 第1章 为什么要学C语言

# 程序设计语言的发展历程

- 百家争鸣,据不完全统计2500 种
  - http://people.ku.edu/~nkinners/LangLis t/Extras/langlist.htm可看到其列表和简 介
- 影响最大、寿命最长的是C语言



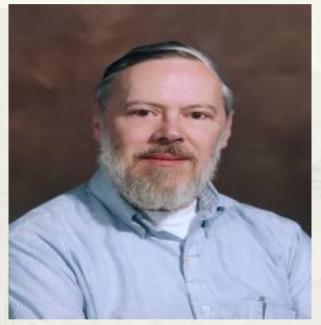
#### 编程语言受欢迎程度排行



## C语言的设计者



Ken Thompson 江湖人称 ken



Dennis M. Ritchie 江湖人称 dmr



Ritchie漫画像

两个骨灰级游戏玩家

#### 无心插柳柳成荫

- 1969年,ken梦想能遨游宇宙,设计了一款模拟在太阳系航行的电子游戏——"Space Travel",1971年dmr加入
- 游戏在PDP-7小型机上开发,因为这台机器免费,但免费的机器没有操作系统
- 玩游戏前,先给PDP-7做个操作系统,命名为UNIX



20世纪60年代,Ritchie 和Thompson 在贝尔实验室开发UNIX操作系统

#### 无心插柳柳成荫

- 汇编编写UNIX太慢,而一般的高级语言 难以实现汇编语言的某些功能,怎么办?
- dmr改造B语言,设计了一种新的高级语言,命名为"C语言",把UNIX的90%以上代码用C改写
- 1983年,因为UNIX和C语言的巨大成功, ken和dmr共同获得当年度的计算机界最 高奖——图灵奖





1999年接受美国国家技术勋章

### C语言标准

- 1978年, Dennis M.Richie和Brian W.Kernighan合著影响深远的名 著《The C Programming Language》
- 1982年,美国国家标准化协会(American National Standards Institute, ANSI)成立C标准委员会
- 1989年,发布第一个完整的C语言标准,简称C89
  - 习惯上也称为ANSIC(标准C)
  - 1990年被国际标准化组织ISO采纳,也称为C90
  - 1999年修正和完善之后,ISO发布了C99

C语言程序设计精髓 7/13

## 为什么学C语言?

- 是一种高级语言
  - \* 并不"高级",只是相对低级语言在一个高的级别上进行编程
  - \* 实际上是一种介于高级语言和低级语言之间的语言,运行效率高
  - \* 透过现象看本质,透过C语言窥探计算机底层原理
- 很多流行语言、新生语言都借鉴了它的思想、语法
  - \* 从C++, 到Java, 再到C#
  - \* 学好C是学习这些流行语言的基础,了解程序设计的基本思想
  - \* 交流、笔试、面试时最常见的语言

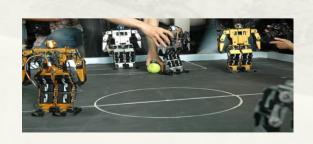
### 为什么学C语言?

- 历史悠久,战勋卓著
  - \* 诞生于20世纪70年代初
  - \* 成熟于80年代
  - \* 很多重量级软件都是用C语言写的,维护已有的C代码
- 上天入地,无所不能
  - \* 几乎没有不能用C语言实现的软件
  - \* 没有不支持C语言的系统

C语言程序设计精髓 9/13

#### C语言适合做什么?

- 选语言的标准——适合的才是最好的
  - \* 编写系统软件(例如OS)和工具软件
  - \* 运行效率要求较高的系统
    - 设备驱动程序,嵌入式系统(Embedded systems)(机器人、 手机和智能家电等)
    - 高性能、实时系统(Real-time systems)





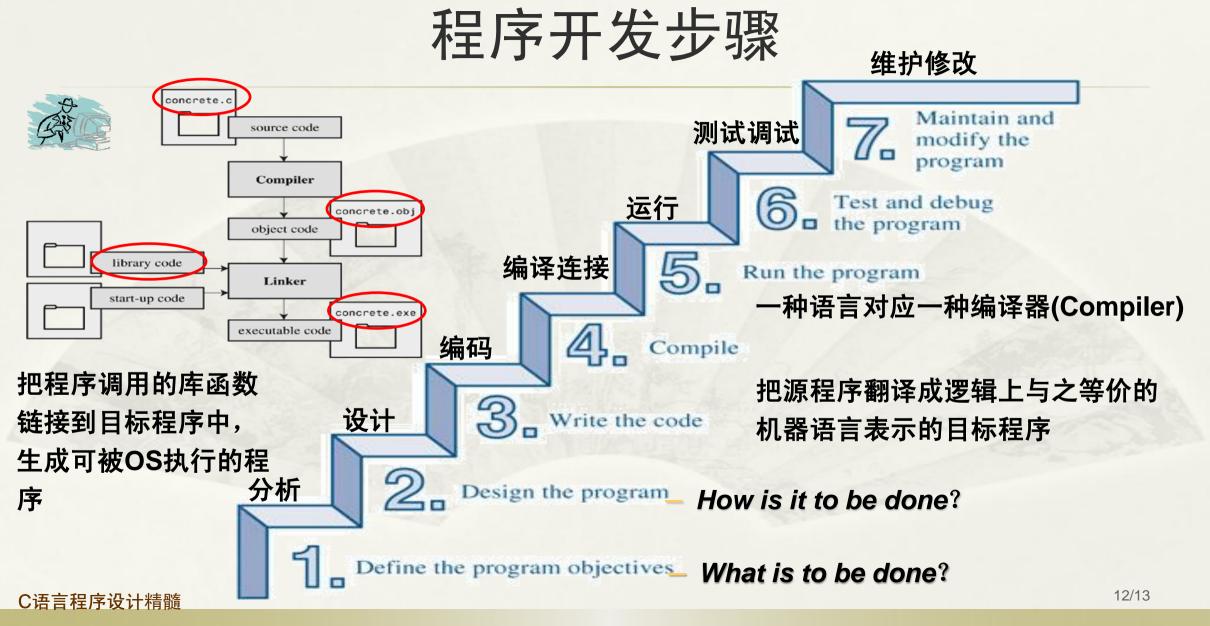






#### C语言适合做什么?

基于 平台的 **Application Domain of** 应用 **High level** 程序 (应用程序) **Domain of** language 应用 平台 **Operation System** 操作系统(OS) **Domain of Low** level language 硬件(Hardware)



#### C程序的开发工具

#### Visual C++

- \* Windows平台上最流行的C/C++集成开发环境之一
- Code::Blocks (简称CB, http://www.codeblocks.org)
  - \* 是近年出现并获得关注的C/C++开发环境
  - \* 免费, 开放源码, 跨平台
  - \* CB只是一个IDE(Integrated Development Environment,集成开发环境),没有内置的编译器和调试器
  - \* 但可支持多种编译器,例如GCC编译器和GDB调试器
  - \* Code::Blocks + GCC (Compiler) + GDB (Debuger)
  - \* http://book.sunner.cn