

python

The Python logo, consisting of two interlocking snakes, one blue and one yellow, is positioned below the word "python".

```
import turtle
turtle.setup(650,350,200,200)
turtle.penup()
turtle.fd(-250)
turtle.pendown()
turtle.pensize(25)
turtle.pencolor("purple")

for i in range(4):
    turtle.circle(40, 80)
    turtle.circle(-40, 80)
    turtle.circle(40, 80/2)
    turtle.fd(40)
    turtle.circle(16, 180)
    turtle.fd(40 * 2/3)
```

Python语言程序设计

程序设计基本方法



python

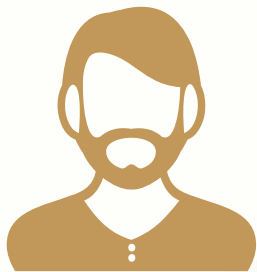
嵩 天
北京理工大学

python



单元开篇

程序设计基本方法



- 计算机与程序设计
- 编译和解释
- 程序的基本编写方法
- 计算机编程





计算机与程序设计

计算机的概念

计算机是根据指令操作数据的设备

- **功能性**

对数据的操作，表现为数据计算、输入输出处理和结果存储等

- **可编程性**

根据一系列指令自动地、可预测地、准确地完成操作者的意图

计算机的发展

计算机的发展参照摩尔定律，表现为指数方式

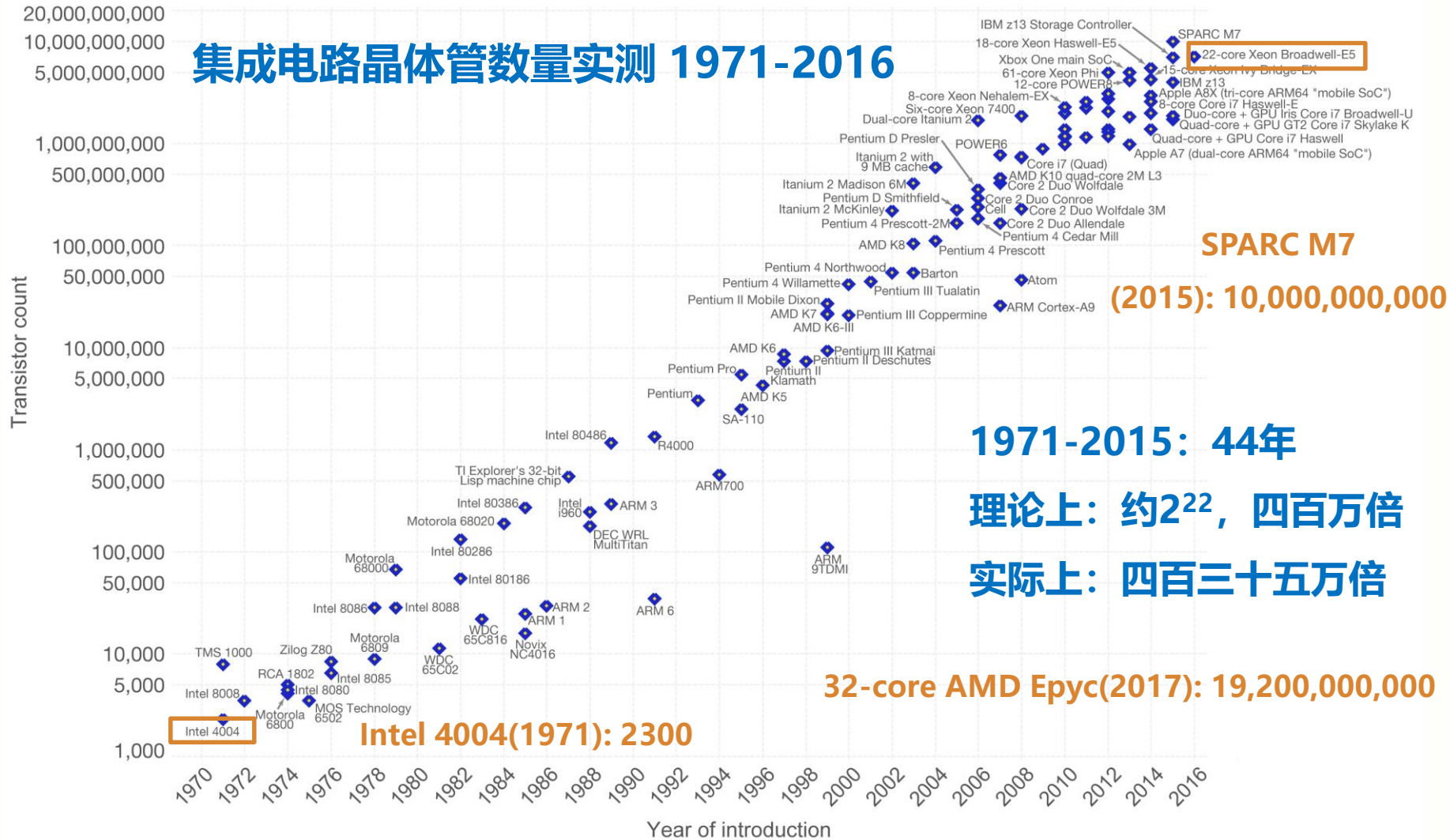
- **计算机硬件所依赖的集成电路规模参照摩尔定律发展**
- **计算机运行速度因此也接近几何级数快速增长**
- **计算机高效支撑的各类运算功能不断丰富发展**

摩尔定律 Moore's Law

计算机发展历史上最重要的预测法则

- Intel公司创始人之一戈登·摩尔在1965年提出
- 单位面积集成电路上可容纳晶体管的数量约每两年翻一番
- CPU/GPU、内存、硬盘、电子产品价格等都遵循摩尔定律

transistor count



计算机的发展

计算机的发展参照摩尔定律，表现为指数方式

- **当今世界，唯一长达50年有效且按照指数发展的技术领域**
- **计算机深刻改变人类社会，甚至可能改变人类本身**
- **可预见的未来30年，摩尔定律还将持续有效**

程序设计

程序设计是计算机可编程性的体现

- **程序设计，亦称编程，深度应用计算机的主要手段**
- **程序设计已经成为当今社会需求量最大的职业技能之一**
- **很多岗位都将被计算机程序接管，程序设计将是生存技能**

程序设计语言

程序设计语言是一种用于交互(交流)的人造语言

- **程序设计语言，亦称编程语言，程序设计的具体实现方式**
- **编程语言相比自然语言更简单、更严谨、更精确**
- **编程语言主要用于人类和计算机之间的交互**

程序设计语言

编程语言种类很多，但生命力强劲的不多

- **编程语言有超过600种，绝大部分都不再被使用**
- **C语言诞生于1972年，它是第一个被广泛使用的编程语言**
- **Python语言诞生于1990年，它是最流行最好用的编程语言**



编译和解释

编程语言的执行方式

计算机执行源程序的两种方式：编译和解释

- **源代码：采用某种编程语言编写的计算机程序，人类可读**

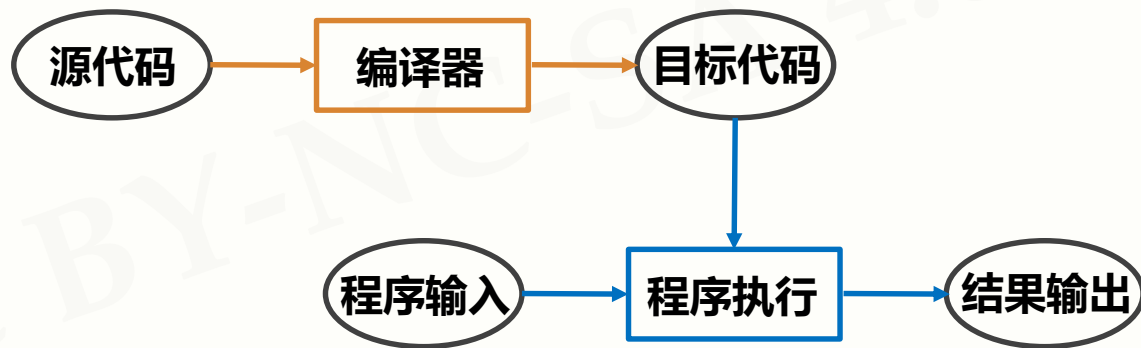
例如：`result = 2 + 3`

- **目标代码：计算机可直接执行，人类不可读 (专家除外)**

例如：`11010010 00111011`

编译

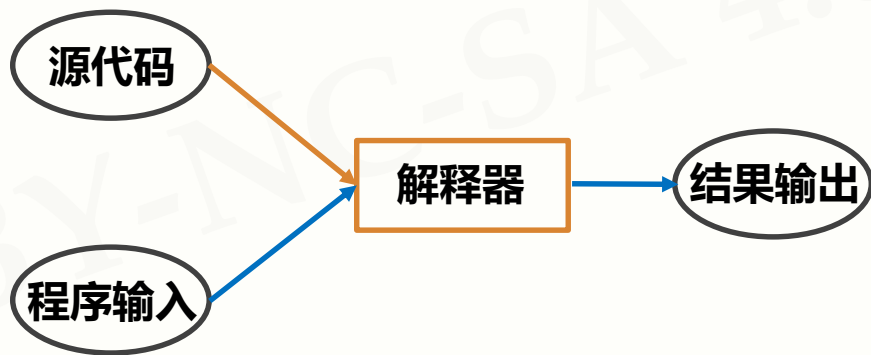
将源代码一次性转换成目标代码的过程



执行编译过程的程序叫作编译器

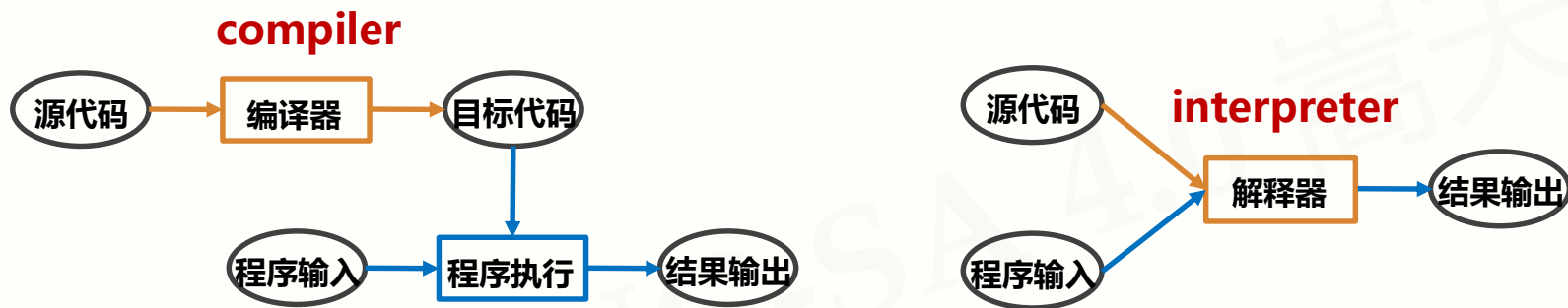
解释

将源代码逐条转换成目标代码同时逐条运行的过程



执行解释过程的程序叫作解释器

编译和解释



- 编译：一次性翻译，之后不再需要源代码（类似英文翻译）
- 解释：每次程序运行时随翻译随执行（类似实时的同声传译）

静态语言和脚本语言

根据执行方式不同，编程语言分为两类

- **静态语言：使用编译执行的编程语言**

C/C++语言、Java语言

- **脚本语言：使用解释执行的编程语言**

Python语言、JavaScript语言、PHP语言

静态语言和脚本语言

执行方式不同，优势各有不同

- **静态语言：编译器一次性生成目标代码，优化更充分**
程序运行速度更快
- **脚本语言：执行程序时需要源代码，维护更灵活**
源代码在维护灵活、跨多个操作系统平台



程序的基本编写方法

IPO

程序的基本编写方法

- **I: Input 输入，程序的输入**
- **P: Process 处理，程序的主要逻辑**
- **O: Output 输出，程序的输出**

理解IPO

输入

- **程序的输入**

文件输入、网络输入、控制台输入、交互界面输入、内部参数输入等

- **输入是一个程序的开始**

理解IPO

输出

- **程序的输出**

控制台输出、图形输出、文件输出、网络输出、操作系统内部变量输出等

- **输出是程序展示运算结果的方式**

理解IPO

处理

- 处理是程序对输入数据进行计算产生输出结果的过程
- 处理方法统称为算法，它是程序最重要的部分
- 算法是一个程序的灵魂

问题的计算部分

一个待解决问题中，可以用程序辅助完成的部分

- **计算机只能解决计算问题，即问题的计算部分**
- **一个问题可能有多种角度理解，产生不同的计算部分**
- **问题的计算部分一般都有输入、处理和输出过程**

编程解决问题的步骤

6个步骤 (1-3)

- 分析问题：分析问题的计算部分，**想清楚**
- 划分边界：划分问题的功能边界，**规划IPO**
- 设计算法：设计问题的求解算法，**关注算法**

使用计算机解决问题


6个步骤 (4-6)

- **编写程序：**编写问题的计算程序，**编程序**
- **调试测试：**调试程序使正确运行，**运行调试**
- **升级维护：**适应问题的升级维护，**更新完善**

求解计算问题的精简步骤

3个精简步骤

- **确定IPO：明确计算部分及功能边界**
- **编写程序：将计算求解的设计变成现实**
- **调试程序：确保程序按照正确逻辑能够正确运行**



计算机编程

Q: 为什么要学习计算机编程?

A: 因为“编程是件很有趣的事儿”!

计算机编程

编程能够训练思维

- 编程体现了一种抽象交互关系、自动化执行的思维模式
- 计算思维：区别逻辑思维和实证思维的第三种思维模式
- 能够促进人类思考，增进观察力和深化对交互关系的理解

计算机编程

编程能够增进认识

- 编程不单纯是求解计算问题
- 不仅要思考解决方法，还要思考用户体验、执行效率等
- 能够帮助程序员加深用户行为以及社会和文化认识

计算机编程

编程能够带来乐趣

- 编程能够提供展示自身思想和能力的舞台
- 让世界增加新的颜色、让自己变得更酷、提升心理满足感
- 在信息空间里思考创新、将创新变为现实

计算机编程

编程能够提高效率

- 能够更好地利用计算机解决问题
- 显著提高工作、生活和学习效率
- 为个人理想实现提供一种借助计算机的高效手段

计算机编程

编程带来就业机会

- 程序员是信息时代最重要的工作岗位之一
- 国内外对程序员岗位的缺口都在百万以上规模
- 计算机已经渗透于各个行业， 就业前景非常广阔

学习编程的误区

Q：编程很难学吗？ A：掌握方法就很容易！

- **首先，掌握编程语言的语法，熟悉基本概念和逻辑**
- **其次，结合计算问题思考程序结构，会使用编程套路**
- **最后，参照案例多练习多实践，学会举一反三**

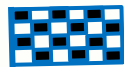
编程辣么好，还等什么？开始学习吧！



单元小结

程序设计基本方法

- 计算机的功能性和可编程性
- 编译和解释、静态语言和脚本语言
- IPO、理解问题的计算部分
- 掌握计算机编程的价值





小花絮

辣么“漂亮”*的学习资料，要不要认识一下制作者？

*仅代表个人审美品味

辣么“清晰”^{*}的内容体系，要不要认识一下设计者？

^{*}仅代表个人审美品味

辣么“精良”*的课程制作，要不要认识一下策划者？

*仅代表个人审美品味

嵩天老师 @北京理工大学

读万卷书 行万里路 只为最好的修炼

微博: weibo.com/songtian425

Email: songtian@bit.edu.cn

学术主页: <http://www.bit-netlab.org/songtian>



