



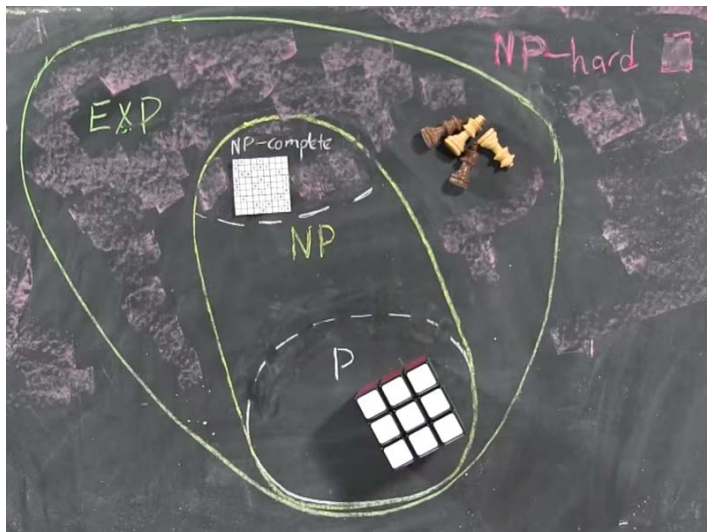
# 数据结构与算法 (Python版)

## 什么是抽象和实现

陈斌 北京大学 [gischen@pku.edu.cn](mailto:gischen@pku.edu.cn)

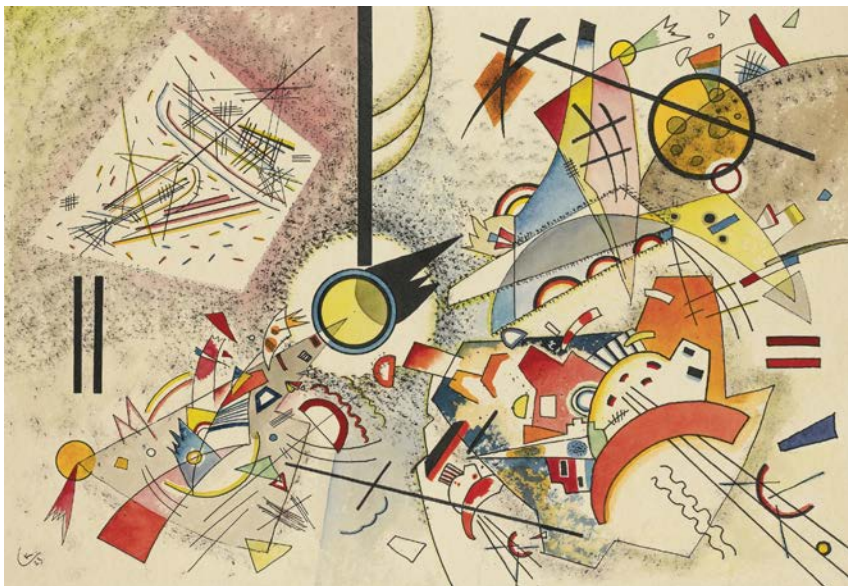
# 计算机科学研究什么

- ❖ 计算机科学不仅仅是对计算机的研究  
虽然计算机是非常重要的计算工具
- ❖ 计算机科学主要研究的是**问题**、**问题解决过程**，以及问题的**解决方案**  
包括了前述的计算复杂性理论  
以及对算法的研究



# 抽象 (Abstraction)

- ❖ 为了更好地处理**机器相关性**或**独立性**，引入了“抽象”的概念
- ❖ 用以从“**逻辑 Logical**”或者“**物理 Physical**”的不同层次上看待问题及解决方案



# 什么是抽象？例子：汽车

- ❖ 从司机观点看来，汽车是一台可以带人去往目的地的代步工具

司机上车、点火、换档、踩油门加速、刹车

- ❖ 从抽象角度说，司机看到汽车的“**逻辑**”层次

司机可以通过操作各个机构来达到运输的目的

- ❖ 这些操纵机构（方向盘、油门、档位）就称为“**接口 Interface**”





# 什么是抽象？ 例子：汽车

- ❖ 而从汽车修理工的角度来看同一辆汽车，就会相当不同，他还需要清楚每项功能是如何实现的  
如发动机工作原理，档位操作的机械结构，发动机舱内各处温度如何测量和控制等等
- ❖ 这些内部构造构成了汽车的“物理”层次
- ❖ 工作过程就称为“实现Implementation”



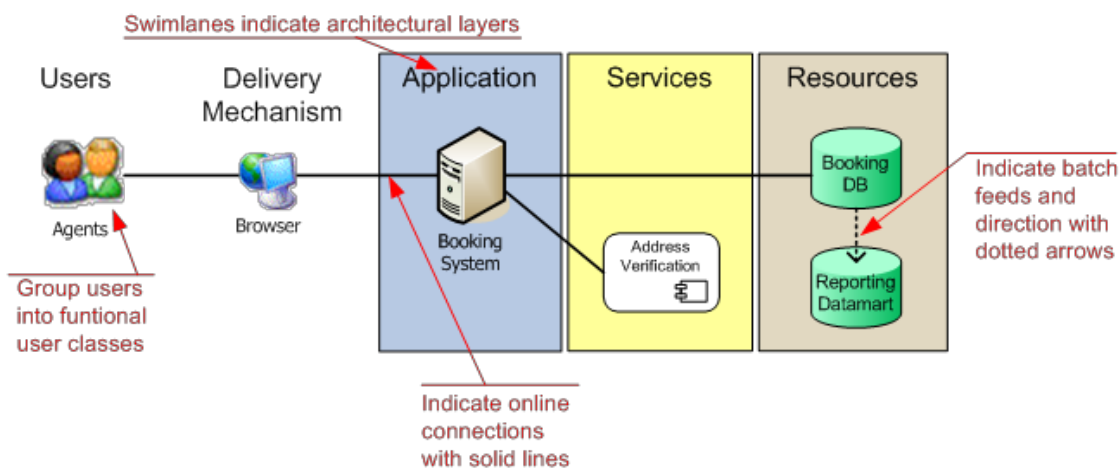
# 什么是抽象？ 例子：计算机

- ❖ 从一般大众用户观点看来，计算机可以用来编辑文档、收发邮件、上网聊天、处理照片等等
- ❖ 并不需要具备计算机内部如何处理的知识  
利用这些功能是计算机的“逻辑”层次



# 什么是抽象？ 例子：计算机

- ❖ 而对于计算机科学家、程序员、技术支持、系统管理员来说
- ❖ 就必须要了解从硬件结构、操作系统原理到网络协议等各方面的低层次细节  
内部如何实现，是计算机的“物理”层次



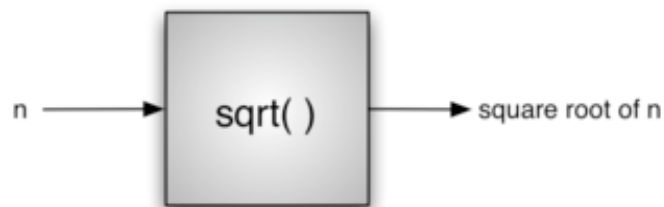
# “抽象”发生在各个不同层次上

❖ 即使对于程序员来说，使用编程语言进行编程，也会涉及到“抽象”

❖ 如计算一个数的平方根

程序员可以调用库函数`math.sqrt`，直接得到结果，而无需关心其内部是如何实现

这种功能上的“黑盒子”称作“过程抽象  
Procedural Abstraction”



```
>>> import math
>>> math.sqrt(16)
4.0
>>>
```



# 抽象与实现：编程

❖ 编程是通过一种程序设计语言，将抽象的**算法**实现为计算机**可以执行的代码**的过程  
没有算法，编程无从谈起

❖ 图灵奖获得者Niklaus Wirth的著名公式：**算法+数据结构=程序**

此公式相当于物理中的 $E=mc^2$

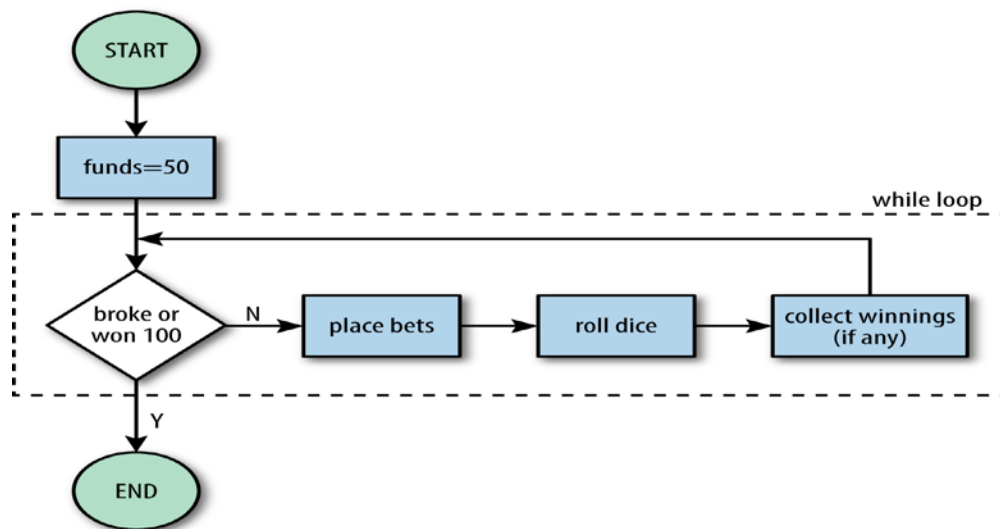
Pascal语言设计者



# 程序设计语言实现算法的基本机制

- ❖ 程序设计语言需要为算法的实现提供实现“**过程**”和“**数据**”的机制  
具体表现为“控制结构”和“数据类型”

- ❖ 程序设计语言均有**语句**对应控制结构  
顺序处理、分支选择、循环迭代



# 程序设计语言实现算法的基本机制

❖ 程序设计语言也提供最基本的**数据类型**来表示数据，如整数、字符等

但对于复杂的问题而言，直接使用这些基本数据类型不利于算法的表达

