





# 1. 程序设计语言

---

- 引言
- 概念
- 特点
- 分类
- 选择要素

# 引言

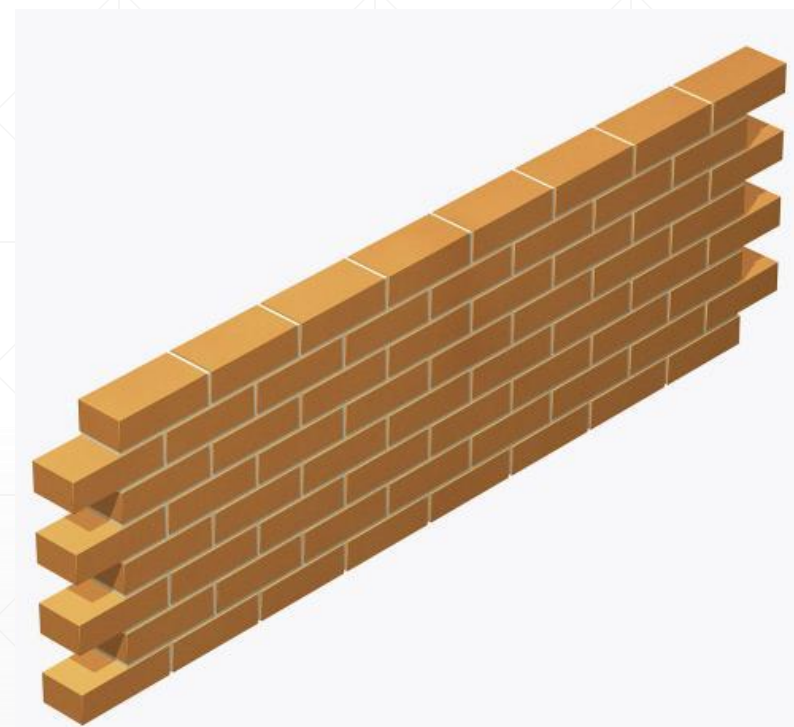
建筑工程中的砌砖：是工程的基础,是设计实现的具体实施。



手艺太差了，还不规范



屋里的墙→砖的应用不合适



手艺不错，规范

# 引言

砖的特点——程序语言特点

砌墙的手艺——编程基本功

砌砖风格——编程风格

砌墙的规范——编程规范

选择合适的程序设计语言——**有效**

良好的编程基本功——**正确**

良好的编程习惯——**易理解、简单、自然**

遵守规范和规则——**可拓展、可伸缩**

× 程序只要通过调试和测试就好，满足需求定义就好

√ 程序应该正确、有效、易理解、简单、自然、可拓展、可伸缩



# 概念—程序设计语言

程序设计语言是：人与计算机通信的最基本工具。

语法+语义+语用



程序语言特性的影响

开发人员的思路和  
解决问题的方式

代码的可理解性、  
可维护性



编码之前的一项重要工作就是选择一种适当的编程语言

## 特点——不同的语言适用于不同的应用

COBOL语言

数据处理程序等

PHP语言

网页处理程序等

C语言

系统软件开发等

JAVA语言

跨平台的应用软件开发等

## 特点——技术方面

需要复杂的  
数据结构

仔细衡量有哪些语言能提供这些复杂的数据结构描述

要求高性能及实时处理能力

选用适合于实时处理的语言，如C或汇编语言

要求许多输出报告  
或繁杂的文件处理

选用PowerBuilder、Delphi或SQL比较合适

# 特点——技术方面

## 程序设计语言 基本成分分类

### 数据成分

指明该语言能接受的数据，如各种类型的变量、数组、指针、记录等

### 运算成分

指明该语言可执行的运算，如 $+$ ， $-$ ， $*$ 、 $/$

### 控制成分

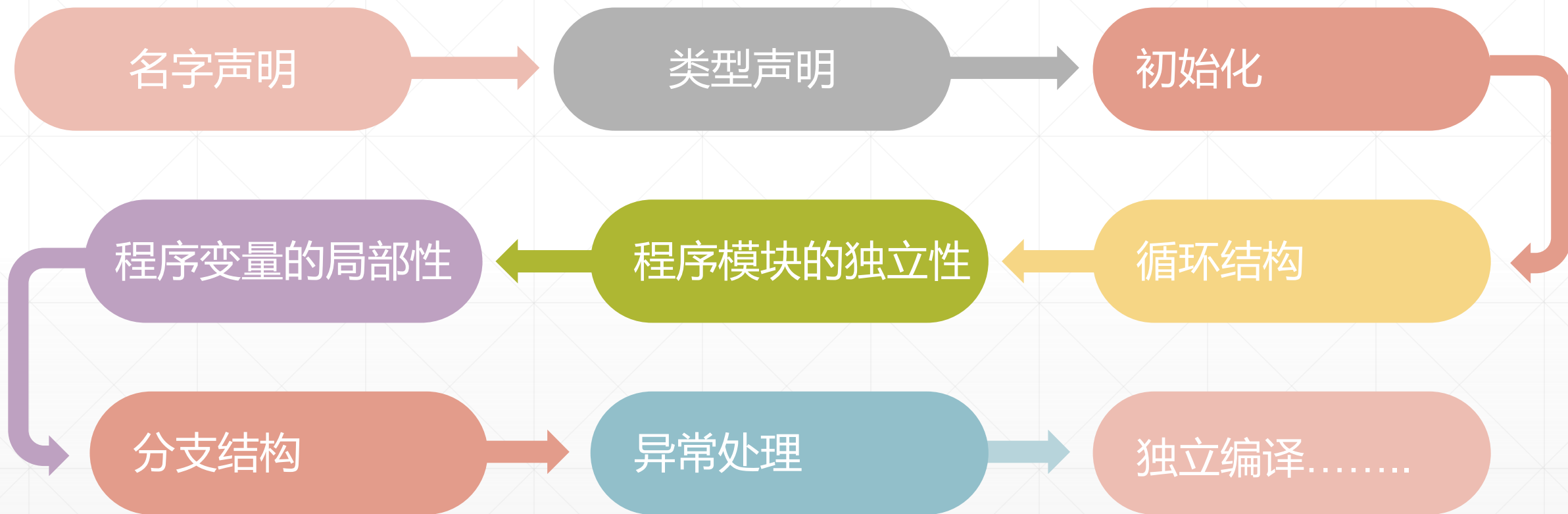
顺序结构、条件选择结构和循环结构

### 传输成分

数据的传输方法，例如输入、输出函数



## 特点——语言本身



# 特点——软件工程方面

## 详细设计

- 能够直接地翻译成程序代码

## 源程序的可移植性

- 应该严格遵守相应的标准编写程序代码

## 编译程序效率较高

- 首先应该支持独立编译，纠错功能强，辅助程序员提高程序调试效率

## 应用代码生成工具

- 许多语言都有与它相应的编译程序、连接程序、调试程序、代码格式化程序、交叉编译程序、宏处理程序和标准子程序库等

## 可维护性

- 源程序的可读性、语言的文档化特性是影响可维护性的重要因素

# 分类

## 语言级别

低级语言（机器语言、汇编语言等）

高级语言（C、Java等）

## 应用范围

通用语言（PASCAL，并发PASCAL, MODULA, XCY, ADA等）

专用语言（BLISS语言专用于PDP-10机器）

## 用户要求

过程式语言（FORTRAN、COBOL、PASCAL、C、Basic等）

非过程式语言（VB、VC、C#等）

## 成分性质

顺序语言（C、COBOL等）

并发语言（JR、ConcurrentC++等）

分布式语言（DCDL、Elang、Scala等）

## 使用方式

交互式语言（BASIC等）、

非交互式语言（FORTRAN、COBOL、ALGOL69、PASCAL、C等）

# 分类——发展历程

- 机器指令代码、不同机器机器语言不同
- 二进制形式
- 使用绝对地址
- 运行效率高、但是出错率高
- 可读性差

第一代语言：  
机器语言

- 符号指令对应于机器指令
- 存储空间由机器安排
- 不同类型处理器具有不同汇编语言

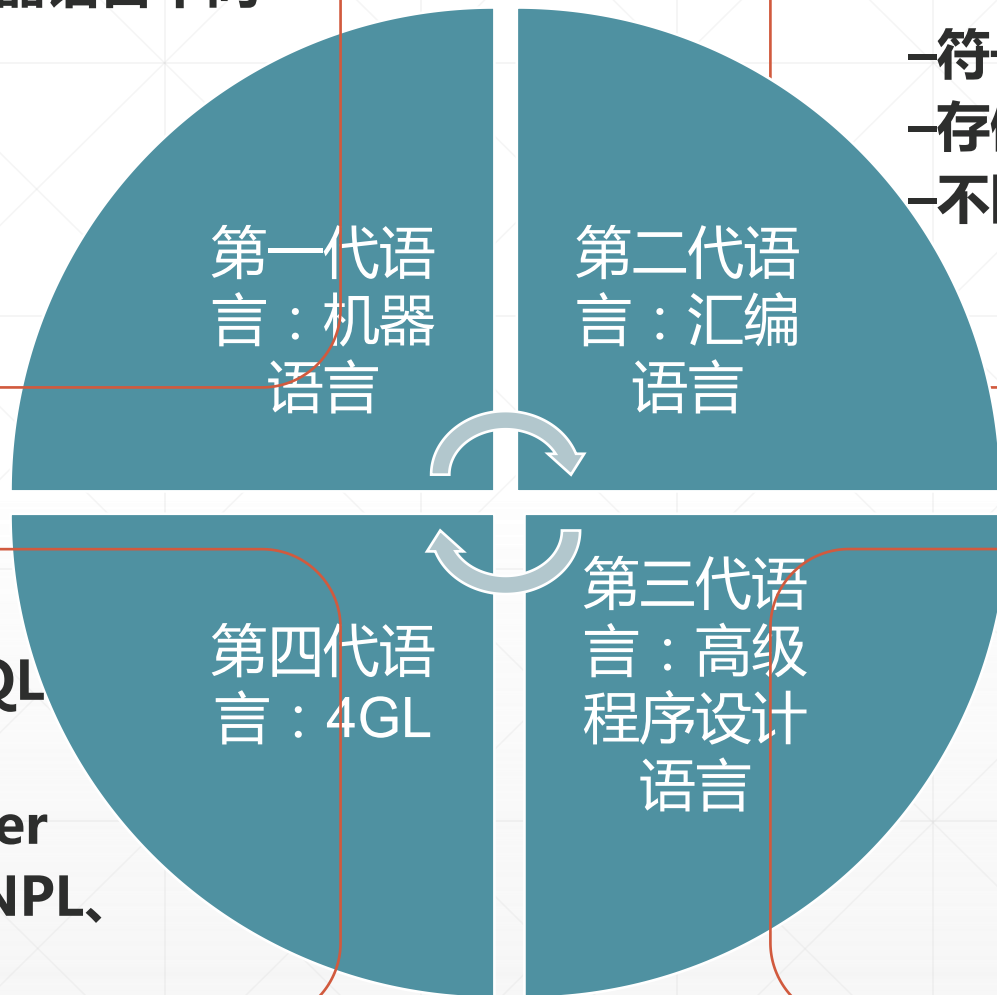
第二代语言：  
汇编语言

- 数据库领域
- 查询语言和报表生成器：SQL
- 图形语言：SQL Windows
- 应用生成器：Power Builder
- 形式化规格说明语言：Z、NPL、SPECINT

第四代语言：  
4GL

- 从20世纪50年代开始
- Fortran、Basic、Pascal、C、C++、C#、Java等

第三代语言：  
高级程序设计语言



# 分类——选择要素

根据应用要求和要求的相对重要性选择

