

嵌入式系统工程师



面向对象程序设计

- 编程语言的发展简史
- 对象、类概念
- 封装、继承、多态特性

1 编程语言的发展简史

1 编程语言的发展简史

➤ 程序设计方法发展历程:

- 机器语言时代
- 汇编语言时代
- 面向过程语言设计

程序=数据+算法

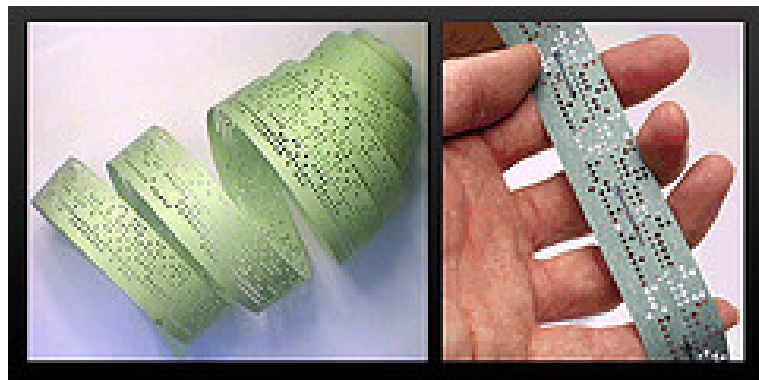
- 面向对象语言设计

程序=（对象+对象+...）

对象=（数据+算法）

如何实现

如何实现



2 对象与类概念

2.1 对象的概念

- 面向对象思想把世界看做是由具有行为的各种对象组成，任何对象都具有某些特征和行为（数据和算法）
- 对象是现实世界中的一个实体（在程序中表现为编译器已经为之分配内存空间的变量）

2.1 对象的概念

► 例如：黎明是一个老师

- 是老师类中的一个对象，对象名：黎明；
- 对象的属性：年龄：30、学历：博士、职称：教授、专业：计算机软件
- 对象的操作：说话、吃饭、授课、科研

► 类的精确定义：具有共性的实体的抽象

- 其本质是自定义数据类型。C语言中的结构体、枚举、联合等也是自定义数据类型。



姓名

年龄

讲课

2.3 类的确定与划分

➤重要性:

- 是软件开发中关键的一步
- 直接影响软件系统的质量
- 直接影响到程序的扩充和代码的重用

➤基本原则:

- 寻求一个大系统中事物的共性, 将具有共性的系统确定为一个类
- 设计类要有一个明确的目标, 一个好的类应该是容易理解和使用的

➤没有统一的标准和固定的方法, 依靠:

- 设计人员的经验、技巧和对问题的把握

3 封装、继承、多态性(设计方法)

3.1 封装性

► 封装性:

对象将私有元素和实现操作的内部细节的结果隐藏起来，并能管理自己的内部状态。外部只能从对象所表示的具体概念、对象提供的服务和对象提供的外部接口来认识对象



3.2 继承性

➤ 继承所表达的是:

- 类之间相关的关系, 这种关系使得对象可以继承另外一类对象的特征和能力

➤ 类之间的继承关系, 具有的特征:

- 类间具有共享特征(包括数据和程序代码的共享)
- 类间具有新增部分(包括非共享的数据和程序代码)
- 类间具有层次结构

➤ 继承的作用:

- 避免公用代码的重复开发, 减少代码和数据冗余
- 通过增强一致性来减少模块间的接口和界面

3.3 多态性

3.3 多态性的概念

- 若语言不支持多态，则不能称之为面向对象的。只支持类而不支持继承和多态的称为基于对象的，如VB
- 多态性是指类中具有相似功能的不同函数使用同一个名称来实现的现象

在C中：

int abs(int)

float fabs(float)

在C++中

int abs(int)

float abs(float)



值得信赖的教育品牌

Tel: 400-705-9680 , Email: edu@sunplusapp.com , BBS: bbs.sunplusedu.com

