目录

[1 抽象数据类型 信息隐藏 2](#_Toc432680624)

[1.1 数据类型 2](#_Toc432680625)

[1.2 抽象数据类型 2](#_Toc432680626)

[1.3 一个例子-Set 2](#_Toc432680627)

[1.4 内存管理 2](#_Toc432680628)

[1.5 对象-Object 2](#_Toc432680629)

[1.6 应用实例 3](#_Toc432680630)

[1.7 一种实现-Set 3](#_Toc432680631)

[1.8 一种实现-Set 3](#_Toc432680632)

[1.9 另种实现-Bag 3](#_Toc432680633)

[1.10 总结 3](#_Toc432680634)

[1.11 练习 3](#_Toc432680635)

[2 动态链接 泛型函数 4](#_Toc432680636)

[2.1 构造器和析构器 4](#_Toc432680637)

[2.2 方法、消息、类和对象 4](#_Toc432680638)

[2.3 选择器、动态链接和多态性 4](#_Toc432680639)

[2.4 应用实例 4](#_Toc432680640)

[2.5 一种实现-String 4](#_Toc432680641)

[2.6 另一种实现-Atom 4](#_Toc432680642)

[2.7 总结 4](#_Toc432680643)

[2.8 练习 4](#_Toc432680644)

[3 编程经验 算术表达式 5](#_Toc432680645)

[3.1 主循环 5](#_Toc432680646)

[3.2 扫描器 5](#_Toc432680647)

[3.3 识别器 5](#_Toc432680648)

[3.4 处理器 5](#_Toc432680649)

[3.5 信息隐藏 5](#_Toc432680650)

[3.6 动态链接 5](#_Toc432680651)

[3.7 后缀添加器 5](#_Toc432680652)

[3.8 算术运算 6](#_Toc432680653)

[3.9 算术运算 6](#_Toc432680654)

[3.10 插入输出 6](#_Toc432680655)

[3.11 总结 6](#_Toc432680656)

[4 继承 代码复用和精炼 7](#_Toc432680657)

1 抽象数据类型 信息隐藏

* 1. 数据类型

对于每种程序设计语言来说，数据类型都是极为主要的部分。ANSI-C就有**int**，**double**

和**char**这些数据类型，而且这只是其中的一小部分。程序员很少满足现状，一种编程语言通常都会提供使用预定义数据类型来构建新类型的方法。一种简单的措施是使用诸如数组、结构体、联合体等聚合成为新数据类型。被托尼·霍尔称为“我们永远无法收回的一步”的指针，允许我们表示和操作本质上极其复杂的数据。

到底什么是数据类型呢？我们可以从几个方面来看。数据类型是值（values）的集合——**char**类型可以表示256种不同的值，**int**类型可以表示更多；他们或多多少类似于数学中的自然数或者整数，是等间隔均匀分布的。**double**类型能表示更多的值，但并不像数学里的实数。

或者我们可以定义一种数据类型，这种类型包括一些值以及作用于这些值的一些操作。通常，这些值是计算机可以表示的，这些操作或多或少反映了可用的硬件指令。从这个方面讲，ANSI-C中的**int**类型做得不是太好：值的集合在不同的机器上可能会有区别，像算术右移位之类的操作也可能表现得不一样。

更复杂的例子

* 1. 抽象数据类型

占位

* 1. 一个例子-Set

占位

* 1. 内存管理

占位

* 1. 对象-Object

占位

* 1. 应用实例

占位

* 1. 一种实现-Set

占位

* 1. 一种实现-Set

占位

* 1. 另种实现-Bag

占位

* 1. 总结

占位

* 1. 练习

2 动态链接 泛型函数

* 1. 构造器和析构器

占位

* 1. 方法、消息、类和对象

占位

* 1. 选择器、动态链接和多态性

占位

* 1. 应用实例

占位

* 1. 一种实现-String

占位

* 1. 另一种实现-Atom

占位

* 1. 总结

占位

* 1. 练习

占位

3 编程经验 算术表达式

占位

* 1. 主循环

占位

* 1. 扫描器

占位

* 1. 识别器

占位

* 1. 处理器

占位

* 1. 信息隐藏

占位

* 1. 动态链接

占位

* 1. 后缀添加器

占位

* 1. 算术运算

占位

* 1. 算术运算

占位

* 1. 插入输出

占位

* 1. 总结

占位

4 继承 代码复用和精炼