

# C++高级编程

## 1. 模板

- 1.1 模板的概念
  - 1.2.1 函数模板的语法
  - 1.2.2 函数模板注意事项
  - 1.2.3 函数模板的案例
  - 1.2.4 普通函数与函数模板的区别
  - 1.2.5 普通函数与函数模板的调用规则
  - 1.2.6 模板的局限性
- 1.2 函数模板
- 1.3 类模板
  - 1.3.1 类模板语法
  - 1.3.2 类模板与函数模板区别
  - 1.3.3 类模板中成员函数创建时机
  - 1.3.4 类模板对象做函数参数
  - 1.3.5 类模板与继承
  - 1.3.6 类模板成员函数类外实现
  - 1.3.7 类模板分文件编写
  - 1.3.8 类模板与友元
  - 1.3.9 类模板案例 (案例比较综合, 含金量比较高)

## 2. STL初识

- 2.1 STL的诞生
- 2.2 STL基本概念
- 2.3 STL六大组件
- 2.4 STL中容器、算法、迭代器
- 2.5 容器算法迭代器初识
  - 2.5.1 vector存放内置数据类型
  - 2.5.2 Vector存放自定义数据类型
  - 2.5.3 Vector容器嵌套容器

## 3 STL- 常用容器

- 3.1 string容器
  - 3.1.1 string基本概念
  - 3.1.2 string构造函数
  - 3.1.3 string赋值操作
  - 3.1.4 string字符串拼接
  - 3.1.5 string查找和替换
  - 3.1.6 string字符串比较
  - 3.1.7 string字符存取
  - 3.1.8 string插入和删除
  - 3.1.9 string子串
- 3.2 vector容器
  - 3.2.1 vector基本概念
  - 3.2.2 vector构造函数
  - 3.2.3 vector赋值操作
  - 3.2.4 vector容量和大小
  - 3.2.5 vector插入和删除
  - 3.2.6 vector数据存取
  - 3.2.7 vector互换容器
  - 3.2.8 vector预留空间
- 3.3 deque容器
  - 3.3.1 deque容器基本概念
  - 3.3.2 deque构造函数
  - 3.3.3 deque赋值操作
  - 3.3.4 deque大小操作
  - 3.3.5 deque 插入和删除
  - 3.3.6 deque 数据存取
  - 3.3.7 deque 排序
- 3.4 案例-评委打分
  - 3.4.1 案例描述
  - 3.4.2 实现步骤
- 3.5 stack容器
  - 3.5.1 stack 基本概念
  - 3.5.2 stack 常用接口
- 3.6 queue 容器
  - 3.6.1 queue 基本概念
  - 3.6.2 queue 常用接口
- 3.7 list容器
  - 3.7.1 list基本概念
  - 3.7.2 list构造函数
  - 3.7.3 list 赋值和交换
  - 3.7.4 list 大小操作
  - 3.7.5 list 插入和删除
  - 3.7.6 list 数据存取
  - 3.7.7 list 反转和排序
  - 3.7.8 排序案例
- 3.8 set/multiset 容器
  - 3.8.1 set基本概念
  - 3.8.2 set构造和赋值
  - 3.8.3 set大小和交换
  - 3.8.4 set插入和删除
  - 3.8.5 set查找和统计
  - 3.8.6 set和multiset区别
  - 3.8.7 pair对组创建
  - 3.8.8 set容器排序
- 3.9 map/multimap容器
  - 3.9.1 map基本概念
  - 3.9.2 map构造和赋值
  - 3.9.3 map大小和交换
  - 3.9.4 map插入和删除
  - 3.9.5 map查找和统计
  - 3.9.6 map容器排序
- 3.10 案例-员工分组
  - 3.10.1 案例描述
  - 3.10.2 实现步骤

## 5 STL- 常用算法

- 5.1 常用遍历算法
  - 5.1.1 for\_each
  - 5.1.2 transform
- 5.2 常用查找算法
  - 5.2.1 find
  - 5.2.2 find\_if
  - 5.2.3 adjacent\_find
  - 5.2.4 binary\_search (二分查找)
  - 5.2.5 count
  - 5.2.6 count\_if
- 5.3 常用排序算法
  - 5.3.1 sort
  - 5.3.2 random\_shuffle
  - 5.3.3 merge
  - 5.3.4 reverse
- 5.4 常用拷贝和替换算法
  - 5.4.1 copy
  - 5.4.2 replace
  - 5.4.3 replace\_if
  - 5.4.4 swap
- 5.5 常用算术生成算法
  - 5.5.1 accumulate
  - 5.5.2 fill
- 5.6 常用集合算法
  - 5.6.1 set\_intersection
  - 5.6.2 set\_union
  - 5.6.3 set\_difference

## 4 STL- 函数对象

- 4.1 函数对象
  - 4.1.1 函数对象概念
  - 4.1.2 函数对象使用
- 4.2 谓词
  - 4.2.1 谓词概念
  - 4.2.2 一元谓词
  - 4.2.3 二元谓词
- 4.3 内建函数对象
  - 4.3.1 内建函数对象意义
  - 4.3.2 算术仿函数
  - 4.3.3 关系仿函数
  - 4.3.4 逻辑仿函数

