广州大学学生实验报告

2018年5月2日

**开课学院及实验室：**计算机科学与工程实验室电子楼412A室

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **学院** | 计算机 | **专业班级** | 软件171 | **姓名** | 谢金宏 | | **学号** | 1706300001 |
| **实验课程名称** | | 面向对象程序设计实验 | | | | **成绩** | |  |
| **实验项目名称** | | 模板 | | | | **指导老师** | | 陶文正 |

# 实验目的及要求

1. 了解函数模板、类模板与模板函数、模板类的关系；
2. 熟悉函数模板、类模板的声明与实例化过程；
3. 掌握应用模板进行的程序设计。

# 实验设备及平台

G++ 7.2.0

# 实验内容及步骤

## 函数模板实验

编写一个适用于求解不同数据类型的多个数值最大值的函数模板，程序如下。

#include <iostream>

**using** **namespace** std**;**

template **<**typename T**>**

T Max**(**T a**,** T b**)**

**{**

**return** a**>**b **?** a **:** b**;**

**}**

template **<**typename T**,** typename**...** Ts**>**

T Max**(**T a**,** Ts**...** b**)**

**{**

T c **=** Max**(**b**...);**

**return** Max**(**a**,** c**);**

**}**

int main**()**

**{**

int a1**,** b1**,** c1**;**

cout **<<** "请输入3个整数" **<<** endl**;**

cin **>>** a1 **>>** b1 **>>** c1**;**

cout **<<** "输入的3个整数为: " **<<** a1 **<<** ' ' **<<** b1 **<<** ' ' **<<** c1**;**

cout **<<** "，最大值 " **<<** Max**(**a1**,** b1**,** c1**)** **<<** endl**;**

long long a2**,** b2**,** c2**;**

cout **<<** "请输入3个长整数" **<<** endl**;**

cin **>>** a2 **>>** b2 **>>** c2**;**

cout **<<** "输入的3个长整数为: " **<<** a2 **<<** ' ' **<<** b2 **<<** ' ' **<<** c2**;**

cout **<<** "，最大值 " **<<** Max**(**a2**,** b2**,** c2**)** **<<** endl**;**

double a3**,** b3**,** c3**;**

cout **<<** "请输入3个浮点数" **<<** endl**;**

cin **>>** a3 **>>** b3 **>>** c3**;**

cout **<<** "输入的3个浮点数为: " **<<** a3 **<<** ' ' **<<** b3 **<<** ' ' **<<** c3**;**

cout **<<** "，最大值 " **<<** Max**(**a3**,** b3**,** c3**)** **<<** endl**;**

**}**

程序的一个可能产生的输出如下：

请输入3个整数

1 2 3

输入的3个整数为: 1 2 3，最大值 3

请输入3个长整数

123 456 789

输入的3个长整数为: 123 456 789，最大值 789

请输入3个浮点数

1.2 3.4 .56

输入的3个浮点数为: 1.2 3.4 0.56，最大值 3.4

## 类模板实验

编写一个类模板，实现一个数组，包括初始化、添加数据、求和、求平均值、显示数组等功能。

#include <iostream>

#include <cassert>

**using** **namespace** std**;**

template **<**typename T**>**

struct Cell **{**

T val**;**

Cell **\*** nxt**;**

Cell**(**T v**,** Cell **\***pre **=** **nullptr)** // 以pre为前驱节点，v为值创建Cell

**{**

**if** **(**pre**)** pre**->**nxt **=** **this;**

val **=** v**;** nxt **=** **nullptr;**

**}**

**};**

template **<**typename T**>**

class Array

**{**

public**:**

Array**(**const T**\*** **=** **nullptr,** int **=** 0**);**

virtual **~**Array**();**

void Append**(**T**);**

T Sum**();**

T Average**();**

void Show**();**

private**:**

Cell**<**T**>** **\***head\_**,** **\***tail\_**;**

**};**

template **<**typename T**>**

Array**<**T**>::**Array**(**const T**\*** gp**,** int len**)** // 利用gp数组的前len个元素初始化

**{**

**if** **(**len **<=** 0**)** len **=** 0**;**

**if** **(**gp **==** **nullptr** **||** len **==** 0**)** // 创建一个空的数组

**{**

head\_ **=** tail\_ **=** **nullptr;**

**return;**

**}**

head\_ **=** tail\_ **=** **new** Cell**<**T**>(**gp**[**0**]);**

**for** **(**int i**=**1**;** i**<**len**;** **++**i**)** **{**

tail\_ **=** **new** Cell**<**T**>(**gp**[**i**],** tail\_**);**

**}**

**}**

template **<**typename T**>**

Array**<**T**>::~**Array**()**

**{**

Cell**<**T**>** **\***cur **=** head\_**;**

**while** **(**cur **!=** **nullptr)**

**{**

Cell**<**T**>** **\***pre **=** cur**;**

cur **=** cur**->**nxt**;**

**delete** pre**;**

**}**

**}**

template **<**typename T**>**

void Array**<**T**>::**Append**(**T val**)**

**{**

**if** **(!**tail\_**)** **{**

head\_ **=** tail\_ **=** **new** Cell**<**T**>(**val**);**

**return;**

**}**

tail\_ **=** **new** Cell**<**T**>(**val**,** tail\_**);**

**}**

template **<**typename T**>**

T Array**<**T**>::**Sum**()**

**{**

Cell**<**T**>** **\***cur **=** head\_**;** T ret **=** 0**;**

**while** **(**cur **!=** **nullptr)** **{**

ret **+=** cur**->**val**;** cur **=** cur**->**nxt**;**

**}**

**return** ret**;**

**}**

template **<**typename T**>**

T Array**<**T**>::**Average**()**

**{**

T ret **=** 0**;** int cnt **=** 0**;**

Cell**<**T**>** **\***cur **=** head\_**;**

**while** **(**cur **!=** **nullptr)** **{**

ret **+=** cur**->**val**;**

**++**cnt**;** cur **=** cur**->**nxt**;**

**}**

**if** **(**cnt **==** 0**)** cerr **<<** "Empty array." **<<** endl**;**

**return** ret**/**cnt**;**

**}**

template **<**typename T**>**

void Array**<**T**>::**Show**()**

**{**

Cell**<**T**>** **\***cur **=** head\_**;**

**while** **(**cur **!=** **nullptr)** **{**

cout **<<** cur**->**val **<<** ' '**;**

cur **=** cur**->**nxt**;**

**}**

cout **<<** endl**;**

**}**

void menu**()**

**{**

cout **<<** "功能菜单> 初始化(1) | 追加(2) | 求和(3) | 求平均数(4) | 显示元素(5) | 退出(0)" **<<** endl**;**

**}**

const int num**[]** **{**2018**,** 5**,** 19**,** 't'**,** 't'**,** 'w'**};**

int main**()**

**{**

cout **<<** "类模板实验二 ~ 静态模板数组" **<<** endl**;**

Array**<**int**>** **\***arr **=** **new** Array**<**int**>(**num**,** 6**);**

int kase **=** 0**;**

**while** **(true)**

**{**

menu**();** cin **>>** kase**;**

**if** **(**kase **==** 0 **||** cin**.**eof**())** **{**

cout **<<** "<清理并退出>" **<<** endl**;**

**break;**

**}**

**if** **(**kase **==** 1**)** **{**

**delete** arr**;** arr **=** **new** Array**<**int**>;**

int count**;**

cout **<<** "请输入元素数量> "**;** cin **>>** count**;**

cout **<<** "请输入数组元素> " **<<** endl**;**

**while** **(**count**--)** **{**

int cur**;** cin **>>** cur**;**

arr**->**Append**(**cur**);**

**}**

cout **<<** "<完成>" **<<** endl**;**

**continue;**

**}**

**if** **(**kase **==** 2**)** **{**

cout **<<** "请输入追加的元素> "**;**

int app**;** cin **>>** app**;**

arr**->**Append**(**app**);**

cout **<<** "<完成>" **<<** endl**;**

**continue;**

**}**

**if** **(**kase **==** 3**)** **{**

cout **<<** "和> " **<<** arr**->**Sum**()** **<<** endl**;**

**continue;**

**}**

**if** **(**kase **==** 4**)** **{**

cout **<<** "平均数> " **<<** arr**->**Average**()** **<<** endl**;**

**continue;**

**}**

**if** **(**kase **==** 5**)** **{**

cout **<<** "显示元素> " **<<** endl**;**

arr**->**Show**();**

**continue;**

**}**

cout **<<** "<无法理解的输入>" **<<** endl**;**

**}**

**delete** arr**;**

**}**

一个可能的运行结果如下：

类模板实验二 ~ 静态模板数组

功能菜单> 初始化(1) | 追加(2) | 求和(3) | 求平均数(4) | 显示元素(5) | 退出(0)

5

显示元素>

2018 5 19 116 116 119

功能菜单> 初始化(1) | 追加(2) | 求和(3) | 求平均数(4) | 显示元素(5) | 退出(0)

1

请输入元素数量> 4

请输入数组元素>

1 2 3 4

<完成>

功能菜单> 初始化(1) | 追加(2) | 求和(3) | 求平均数(4) | 显示元素(5) | 退出(0)

2

请输入追加的元素> 5

<完成>

功能菜单> 初始化(1) | 追加(2) | 求和(3) | 求平均数(4) | 显示元素(5) | 退出(0)

5

显示元素>

1 2 3 4 5

功能菜单> 初始化(1) | 追加(2) | 求和(3) | 求平均数(4) | 显示元素(5) | 退出(0)

3

和> 15

功能菜单> 初始化(1) | 追加(2) | 求和(3) | 求平均数(4) | 显示元素(5) | 退出(0)

4

平均数> 3

功能菜单> 初始化(1) | 追加(2) | 求和(3) | 求平均数(4) | 显示元素(5) | 退出(0)

0

<清理并退出>

# 总结

C++的模板能帮助程序员简化代码，常用的模板类型有函数模板和类模板。常用的模板语法并不复杂，多加练习就能掌握；有些不常用的模板语法，如C++11中的新特性模板参数包、模板显式实例化，模板类的友元等等，注意思考实现的原理，需要用到时也不难回忆起来。