

OOP1 A Matrix Class Template Which Supports Linear Operation

一个支持线性运算的矩阵类

卢思迪

2016 年 10 月 15 日

目录

1	一些想法	3
1.1	为什么要使用类模板而不是最普通的类	3
1.2	这种写法相较于写成普通类的好处	3
2	目标	3
3	成员说明	3
3.1	成员变量	3
3.2	成员函数及友元函数	3
3.2.1	matrix()	3
3.2.2	matrix(const matrix & src)	4
3.2.3	matrix & operator=(const matrix & src)	4
3.2.4	type_real & operator()(size_t i, size_t j) 及其 const 版本	4
3.2.5	下标运算符	4
3.2.6	size_t size() const	4
3.2.7	加法运算符和减法运算符	4
3.2.8	基于流的输入输出运算符	4
3.2.9	析构函数	4
4	非强制性 Bonus 目标 (选做, 没有额外加分)	4
4.1	没有加分为什么要选做这个	4
4.2	实现矩阵乘法和求逆	5
4.3	求矩阵的特征值和特征向量	5

4.4	求矩阵的范数	5
4.5	5

1 一些想法

1.1 为什么要使用类模板而不是最普通的类

因为在接下来的很多应用情境（尤其是第二学期的数据结构课上）里，类模板实际上出现得比普通的类要多，而模板与元编程在本学期书的最后一章，有可能进度会来不及讲，因此为了让大家到时候不至于慌乱，我试图用一个非常简单，简单到几乎不能算是模板的类模板来给大家一个缓冲的时间。

1.2 这种写法相较于写成普通类的好处

我们可以控制矩阵中每一个元素的精度，这样这个矩阵类完成时，根据需要他可以是定义在整数域上的矩阵，也可以是定义在实数域上的矩阵，而且我们还可以使用我们自己写的高精度浮点数/有理数/可计算数类来代替 `double` 来进一步提高精度到任意我们满意的程度。

2 目标

实现一个支持线性运算的方阵类（模板）。它的大小在实例化时确定，不可以中途修改。此外不允许使用静态数组成员管理内存。不允许使用 STL。

提示：本文中出现的所有术语均可以在翁阿姨的书上找到定义或诠释，请仔细阅读教材！

如果有任何其他问题，请联系助教：卢思迪 (aka 兔君 =。=(289009337))

3 成员说明

3.1 成员变量

要求使用私有变量来保证封装性。为了锻炼同学们管理内存的能力，不允许使用 STL 及 c-style 数组成员来管理内存，应该使用指针 +new 或者 `malloc` 的方式开空间。

3.2 成员函数及友元函数

3.2.1 `matrix()`

这是 `matrix` 类的构造函数。你应该在这里进行内存的分配、初始化等工作。

3.2.2 `matrix(const matrix & src)`

这是 `matrix` 类的拷贝构造函数。你应该在这里实现一个好的**深拷贝**的构造函数，使得对于 `matrix` 类的拷贝构造能够**安全地**完成，而不是拷贝源被析构后得到的副本也一并失效。

3.2.3 `matrix & operator=(const matrix & src)`

这是 `matrix` 类的赋值运算符。你应该在这里实现一个好的**深拷贝**的赋值运算符，并且正确地返回一个**本对象引用**。

3.2.4 `type_real & operator()(size_t i, size_t j)` 及其 `const` 版本

通过类似 `mat(x, y)` 的方式来访问矩阵中第 `x` 行第 `y` 列的元素。1-based 下标。

3.2.5 下标运算符

需要支持使用类似 `mat[x][y]` 的方式来访问矩阵中第 `x` 行第 `y` 列的元素。同样要求有 `const` 和非 `const` 两个版本。1-based 下标。

3.2.6 `size_t size() const`

返回方阵的阶。

3.2.7 加法运算符和减法运算符

需要支持加法运算符和减法运算符，来实现类似 `mat1 + mat2` 和 `mat1 - mat2` 的语法。使用成员函数或者友元函数均可。

3.2.8 基于流的输入输出运算符

支持使用类似 `cin >> mat` 和 `cout << mat` 的语法来对矩阵进行输入输出。将一个矩阵输出后的内容作为输入得到的矩阵应该是原矩阵。

3.2.9 析构函数

正确实现析构函数，不得出现**内存泄漏**。

4 非强制性 Bonus 目标（选做，没有额外加分）

4.1 没有加分为什么要选做这个

你的线性代数和科学计算等课程会用到的。【现在不做将来还是要做 =。=

- 4.2 实现矩阵乘法和求逆
- 4.3 求矩阵的特征值和特征向量
- 4.4 求矩阵的范数
- 4.5