C++ 课前热身练习实验报告

袁雨 PB20151804

一、实验目的

- 学习使用 CMake 来搭建项目
- 学习使用 Visual Studio 2019 进行编程, 学会其 debug 工具来调试代码
- 学习面向对象 C++ 编程,特别是类(class) 的封装特性及构造函数、析构函数、函数重载、运算符重载等
- 熟悉 C++ 指针、动态内存分配、预编译头机制等
- 学习模板 template
- 学习STL的 vector、list、map 等
- 学习静态库 lib, 动态库 dll 的编写

二、实验内容

小练习 1. 基础的动态数组

详细说明见于 <u>documents/1 BasicDArray</u>

完成 src/executables/1 BasicDArray

小练习 2. 高效的动态数组

详细说明文档见于 documents/2 EfficientDArray

完成 src/executables/2 EfficientDArray

小练习 3. 模板动态数组

详细说明见于 documents/3 TemplateDArray

仿照小练习 1,在文件夹 src/executables/ 中添加文件夹 3_TemplateDArray ,并在其内

- 添加文件 TemplateDArray.h
- 添加文件 main.cpp
- 添加文件 CMakeLists.txt ,同于 <u>src/executables/1 BasicDArray/CMakeLists.txt</u>

重新 CMake 后得到新子项目 3_TemplateDArray

小练习 4. 基于 list 的多项式类

详细说明见于 documents/4 list Polynomial

这里将 PolynomialList 编写成了动态库,具体查看 <u>src/libraries/shared</u>,编译后会生成 <u>lib/</u>CppPractices_libraries_shared(d).dll

你需要补充完成 src/libraries/shared/PolynomialList.cpp

小练习 5. 基于 map 的多项式类

详细说明见于 documents/5 map Polynomial

这里将 PolynomialMap 编写成了静态库,具体查看 <u>src/libraries/static</u>,编译后会生成 <u>lib/</u>CppPractices_libraries_static(d).dll

你需要补充完成 src/libraries/static/PolynomialMap.cpp

5 map Polynomial 会测试该静态库,另外该子项目还用到了小练习 4 的动态库 PolynomialList,其中会测试小练习 4 和小练习 5 的性能差异

三、实验结果

小练习 1. 基础的动态数组

```
Microsoft Visual Studio 调试控制台
                                                                            X
elements: 2.1
elements: 2.1 3 3.1 3.2
elements: 3 3.1 3.2
elements: 4.1 3 3.1 3.2
elements: 4.1 3 3.1 3.2
elements: 4.1 3 3.1 3.2
size: 4
elements: 4.1 3 3.1 3.2
elements: 21
elements:
elements: 22 0 0 0 0
elements: 100 97 98 99
D:\桌面文件\数学建模\USTC_CG-master\Homeworks\0_CppPratices\project\bin\
CppPractices_executables_1_BasicDArrayd.exe (进程 23172)已退出,代码为 0
按任意键关闭此窗口. . .
```

小练习 2. 高效的动态数组

```
Microsoft Visual Studio 调试控制台
                                                                 П
                                                                       X
elements: 2.1 3 3.1 3.2
size: 3
elements: 3 3.1 3.2
size: 4
elements: 4.1 3 3.1 3.2
size: 4
elements: 4.1 3 3.1 3.2
size: 4
elements: 4.1 3 3.1 3.2
elements: 4.1 3 3.1 3.2
size: 4
elements: 4.1 3 3.1 3.2
size: 1
elements: 21
size: 0
elements:
size: 5
elements: 22 0 0 0 0
size: 4
elements: 100 97 98 99
D:\桌面文件\数学建模\USTC_CG-master\Homeworks\0_CppPratices\project\bin\
CppPractices_executables_2_EfficientDArrayd.exe (进程 27324)已退出,代码
为 0。
要在调试停止时自动关闭控制台,请启用"工具"->"选项"->"调试"->"调试
停止时自动关闭控制台"。
按任意键关闭此窗口...<u>-</u>
```

小练习 3. 模板动态数组

```
™ Microsoft Visual Studio 调试控制台
                                                                X
elements: 2.1
size: 4
elements: 2.1 3 3.1 3.2
size: 3
elements: 3 3.1 3.2
size: 4
elements: 4.1 3 3.1 3.2
size: 1
elements: 21
size: 0
elements:
size: 5
elements: 22 0 0 0 0
size: 4
elements: d a b c
D:\桌面文件\数学建模\USTC_CG-master\Homeworks\O_CppPratices\project\bin\
CppPractices_executables_3_TemplateDArrayd.exe (进程 2252)已退出,代码为
按任意键关闭此窗口. . . _
            aconv3 Print()
```

小练习 4. 基于 list 的多项式类

```
Microsoft Visual Studio 调试控制台

2x^1 -3x^4
4x^1 -3x^4
6x^1 -6x^4
-2x^1
8x^2 -18x^5 +9x^8

D:\桌面文件\数学建模\USTC_CG-master\Homeworks\0_CppPratices\project\bin\CppPractices_executables_4_list_Polynomiald.exe(进程 22972)已退出,代码为 0。
按任意键关闭此窗口...
```

小练习 5. 基于 map 的多项式类

```
Test Map:

48x 164 + 22x 1549 + 43x 1677 + 40x 7030 + 71x 8953

48x 195 + 37x 490 + 92x 1001 + 1x 1840 + 88x 4086

1466x 359 + 1813x 654 + 44508x 1165 + 748x 1744 + 1462x 1872 + 49x 2004 + 814x 2039 + 1591x 2167 + 2024x 2550 + 3956x 2678 + 22x 3389 + 43x 3517 + 4312x 4250 + 1936x 5635 + 3784x 5763 + 1360x 7225 + 1480x 7520 + 3680x 8031 + 40x 8870 + 2414x 9148 + 2627x 9443 + 6532x 9954 + 71x 10793 + 3520x 11116 + 6248x 13039

49x 164 -34x 195 -37x 490 + 92x 1001 + 22x 1549 + 43x 1677 + 1x 1840 + 88x 4086 + 40x 7030 + 71x 8953

1est Constructor time: 51

Test Map:

49x 164 -34x 195 -37x 490 + 92x 1001 + 1x 1840 + 88x 4086

19x 164 -34x 195 + 37x 490 + 92x 1001 + 1x 1840 + 88x 4086

19x 164 -34x 195 -37x 490 + 92x 1001 + 1x 1840 + 88x 4086

19x 164 -342x 1549 + 43x 1677 + 40x 7030 + 71x 8953

34x 195 -37x 490 + 92x 1001 + 1x 1840 + 88x 4086

19x 164 -342x 1549 + 43x 1677 + 40x 7030 + 71x 8953

34x 195 -37x 490 + 92x 1001 + 1x 1840 + 88x 4086

19x 164 -34x 195 -37x 490 + 92x 1001 + 1x 1840 + 88x 4086

19x 164 -34x 195 + 37x 490 + 92x 1001 + 1x 1840 + 88x 4086

19x 164 -34x 195 + 37x 490 + 92x 1001 + 22x 1549 + 43x 1677 + 1x 1840 + 88x 4086 + 40x 7030 + 71x 8953

19x 164 -34x 195 + 37x 490 + 92x 1001 + 22x 1549 + 43x 1677 + 1x 1840 + 88x 4086 + 40x 7030 + 71x 8953

19x 164 -34x 195 + 37x 490 + 92x 1001 + 22x 1549 + 43x 1677 + 1x 1840 + 88x 4086 + 40x 7030 - 71x 8953

19x 164 -34x 195 +37x 490 + 92x 1001 + 22x 1549 + 43x 1677 + 1x 1840 + 88x 4086 + 40x 7030 - 71x 8953

19x 164 -34x 195 +37x 490 + 92x 1001 + 22x 1549 + 43x 1677 + 1x 1840 + 88x 4086 + 40x 7030 - 71x 8953

19x 164 -34x 195 +37x 490 + 92x 1001 + 22x 1549 + 43x 1677 + 1x 1840 + 88x 4086 + 40x 7030 - 71x 8953

19x 164 -34x 195 +37x 490 + 92x 1001 + 22x 1549 + 43x 1677 + 1x 1840 + 88x 4086 + 40x 7030 - 71x 8953

19x 164 -34x 195 +37x 490 + 92x 1001 + 22x 1549 + 43x 1677 + 1x 1840 + 88x 4086 + 40x 7030 - 71x 8953

19x 164 -34x 195 + 37x 490 + 92x 1001 + 22x 1549 + 43x 1677 + 1x 1840 + 88x 4086 + 40x 7030 + 71x 803 + 71x 1079 + 71x 1079 + 71x 1079 + 71x 1079
```

四、实验分析与心得体会

通过小练习1,了解和掌握了 C++ 的基本语法。初步了解类(对象)的编写,了解构造函数、析构函数、函数重载等;熟悉和巩固了指针、动态内存分配的机制与操作。

通过小练习2,学会了使用预先多分配一些内存的方法来提高动态数组的效率,理解对象的public接口的重要性。

通过小练习3, 学会了使用 template 来处理不同类型的动态数组类, 初步了解和使用 STL 的 vector。

通过小练习4, 学会了使用 vector 和 list、了解生成和使用动态库。

通过小练习5, 学会了使用 map 来实现 polynomial 类, 了解静态库的编写与使用;

对比发现,list<r> 容器是双向链表,因此可以有效的在任何位置添加和删除。列表的缺点是不能随机访问内容,要想访问内容必须在列表的内部从头开始便利内容,或者从尾部开始。 map<k,r> 映射容器:K表示键,T表示对象,根据特定的键映射到对象,可以进行快速的检索。 map 的效率更高,二者的时间复杂度不同。