

**ICT283实验室8练习（本实验室不考核，但必须完成）。****目标：**

- 完成并赶上所有过去的实践工作
- 了解做OO重新设计时的的问题
- 学习使用STL矢量
- 学习队列数据结构
- 学习堆栈数据结构的用途
- 要做测试
- 为作业2做准备 - 下面的问题4必须现在完成，因为它是作业2的要求之一。对这个问题的评估是作业2的一部分。你需要提交问题4，以获得后面的材料。

不落下这些练习是非常重要的。

**你应该注意，即使一个练习不被评估，但不尝试该练习将使你很难理解后续材料。如果你不完成实验，你的作业2也会有问题。**

如果你想在自己的电脑上工作，先安装graphviz，然后安装doxygen。

**锻炼身体**

在你弄清楚所需的确切内容之前，**不要**开始编码。在纸上完成。画一个UML图，说明类是如何连接和使用的。确保你用doxygen风格的注释来记录所有代码。

**想一想你将用什么测试数据来证明你的程序是有效的。你将如何写一个测试计划？**见主题8的讲义。以后的单元所需的测试（文件夹“*三年级单元的测试*”）提供了所有IT学生必须完成的最后项目单元的例子。对于这个单元，使用主题8的讲座材料。在“*三年级单元的测试*”中的样本将给你一个想法，你将在三年级的预期。

1. 如果你在过去的实践工作中落后了，请在本周补上。

转换你为作业1所写的模板Vector类的内部表示，使你的Vector使用STL<sup>1</sup> 矢量。你的*Vector.h*将有`#include <vector>`。这种方法类似于讲义中的Queue类，我们自己的Queue类封装了STL队列。主题7有一个类似于Stack的例子。在对你的*Vector*类的*Vector.h*进行修改之前，先看一下Stack和Queue的例子。

这是一种 "组合 "关系，你的Vector封装了一个STL向量，这在作业1中是不允许的，但现在你可以在你自己的Vector中封装时使用STL向量了。如果你在你的作业1中使用这个新的Vector，你的作业1是否需要修改代码？试试吧，看看。如果作业（应用端--主程序）需要修改代码，那么你的Vector就没有完成好。客户端程序应该只依赖于你的Vector的公共接口。

唯一需要改变的是在你的Vector类的方法主体和私有数据中使用STL向量。你的客户代码（主程序和其他代码）不应该有任何变化，因为它依赖于你的Vector。如果客户端代码有变化，那么你就知道你的Vector没有很好的设计任务。记下出现的变化问题。你可能想和你的导师讨论这个问题。你能用你的新Vector来存储其他对象吗？运行这个新的

---

<sup>1</sup>如果你在你的作业1中实际使用了STL向量，那么你就违反了作业要求。下次要更仔细地阅读规范。你也应该注意一下问题和答案文件。

通过实验5和作业1所使用的相同单元测试的Vector。你的Vector是否可以作为引用和常量引用传递给其他例程，并在这些例程中得到利用？

2. 在你完成上述所有练习后再做这个练习。实施Lec- 23 Animation.ppt中的计算器例子。为了便于测试，从一个文件中读取表达式，每个表达式只有一行。如果你的时间不够，可以先做第4题（如下），然后再来做本题。
3. 检查Queue的实现。是否还需要什么方法？改变示例的实现，以便不使用STL数据结构（或算法）。类的接口不能改变，QueueTest的程序代码不能被触动，但仍应正确构建和运行。
4. 在这个练习中，你将扩展作业1以读取多个数据文件。你将继续使用你在作业1中使用的同一个Vector。你将需要能够为作业2读取多个数据文件，所以现在需要解决多文件读取问题。运行作业1，多个文件被用来解决菜单选项。你在作业2中需要的文件已经在数据文件夹中。

使用以下方法：

- a. 使用至少4个不同年份的数据文件。将所有4个文件放在一个名为data的子目录中。你不需要需要下载任何更多的数据，因为数据文件夹中已经提供了足够的样本数据。
- b. 将CSV数据文件的文件名复制到一个名为data\_source.txt的文本文件中。你的程序将首先打开data/data\_source，从data\_source.txt中读取每个CSV数据文件名，并将CSV数据文件的内容加载到你的作业1中。你不能假设CSV数据文件在data\_source.txt中以任何排序顺序列出。任务2所需的高水平算法，你现在应该实现：

---

```
打开 数据/ data_source.txt
```

```
对于data/data_source.txt中列出的每个CSV文件
```

```
    打开CSV文件，将数据读入你在作业1中使用的数据结构中。
```

```
    读取完成后，关闭CSV数据文件 Endfor
```

```
显示菜单
```

---

在菜单显示给终端用户之前，程序会读取所有文件并加载适当的数据结构。

菜单选项现在应该可以用于几个不同的年份。

在data\_source.txt中添加更多的数据文件，测试你的作业1。你必须在数据结构变化被纳入作业2之前，现在就完成这个任务。

**请注意：**

- 你应该用更多的数据文件来测试你在问题4中的程序。问题4和作业2需要同时处理多个数据文件，并有各种列（字段）的安排。你不需要下载更多的数据，因为已经提供了足够的样本数据。如果你想的话，可以从<http://www.met.murdoch.edu.au/> 下载。下载时，使用左边的数据组，不要勾选UTC日期/时间，并在网站上选择所需的数据列。不要选择降雨量（10分钟）和蒸发量（10分钟）。你将得到一个.csv文件，该文件的顶部会有传感器的密钥。
  - 如果你自己下载的数据，在Microsoft Excel中打开csv文件，删除最上面的几行，但保留标题行。检查Excel中的数据/时间格式是否正确，以及

然后将该文件保存为本实验室数据目录下的.csv文件。如果日期不是`dd/mm/yyyy`的格式，请使用Excel的帮助来找出如何改变日期格式，然后按照前面的方法保存。检查时间值是否合理。你可以通过检查在珀斯应该是黑暗（没有太阳）的时候不应该有大的太阳辐射值记录。如果你看到天黑时记录的数值很大，那么时间就不是西澳时间（WAST），而是UTC（如GMT--落后8小时）。在notepad++ (<https://notepad-plus-plus.org/>) 中打开csv文件，检查你的程序在读入文件时将看到什么。

- 你现在需要完成第4题，因为你需要提交它来获取后面的材料。本题中正在实现的算法是作业2的一部分。
- 必须进行应用测试。该电子表格给出了一个例子。
  - 请不要伪造/篡改任何测试结果。如果评分员发现任何假的或伪造的结果，将被判处0分。
  - 更详细的例子可以在 "三年级单元测试" 文件夹中找到。在这个二年级单元中，不需要这样的细节水平。

5. 一旦你把上面的问题4解决了，把作业中的Vector改成上面问题1中使用的那个。测试以确保一切工作如常。注意，除了#include问题1的Vector而不是作业1的Vector之外，你不应该对代码进行修改。