

LAB 5

助教：钱宜嘉(19302010065@fudan.edu.cn)

背景介绍

在本次 Lab 中，需要模拟全家零售店售卖系统，每天早上进货，进行销售，晚上清理过期商品，不断重复。

需求

1. 我们至少需要两个类，一个是 FamilyMart 类，一个是商品类。根据自己的逻辑需求定义相关函数以及属性，但商品类必须包含以下属性：

- 名称
- 价格
- 有效期
- 生产日期

注：

- 不同种类的商品名称不同
- 同一种商品有相同的价格和**可能不同**的有效期，有效期单位默认为天
- 每一件商品可以有不同的生产日期

2. 需要管理一个全家商店，每天早上你会得到一份今天需要购进的商品清单。假设所有产品你已经买到，在 [purchase.txt](#) 中。当运行你的商店时，你可以出售你的商品。我们会给你一个销售清单 [sell.txt](#)。
3. 你的 FamilyMart 可能以折扣价出售一些商品，具体折扣在 `sell.txt` 中指出。
4. 回答问题：
重载和默认参数之间的区别？
你是如何实现你的折扣函数？重载，默认参数还是其他？为什么？
5. 在运行一天后，应该检查商品是否过期。如果一个商品已经过期，就丢掉它。可以认为生产日期是生产当天非常早的时候生产出的商品，检查商品的时间是当天晚上很晚的时间，当天过期的商品需要扔掉。例如 5.2 生产的商品，有效期 3 天，则 5.4 的晚上需要扔掉。
6. 应该出售给顾客最先过期（但还没有过期）的商品，以便有更好的收益，但是不能调整顾客的购买顺序，出售必须按 `sell.txt` 中的顺序进行。
7. 在任何情况下都不允许卖给顾客过期的商品。
8. 你可以灵活创建自己的类文件，声明和实现自己的函数，但必须实现 [FamilyMart.h](#) 中提供的接口。[main.cpp](#) 是助教测试程序正确性的示例，但这不是助教进行正确性测试的最终版本，你可以自行修改以便于在本地测试。
9. 默认**第一次开店时间**为 2022 年 5 月 2 日，你可以使用助教提供的 `purchase.txt` 和 `sell.txt` 进行测试。若需要更多测试用例，可以自己生成，但需要保证助教提供的两个文件可以正常测试。最终测试的测试样本和提供样本的格式相同。
10. 附加分：尝试实现性能更好的程序，并在文档中说明具体实现，可以用表格，图片等形式体现改善效果。具体角度例如：有没有方法不遍历所有商品，就找到最早过期的某个商品，其他任何对性能有提升的切入角度均可。不要求达到最佳性能，不支持不鼓励在性能方面内卷，只要有一定的优化提升，并在文档中说明，都可以获得酌情加分。

提交

截止日期：北京时间 2022 年 5 月 8 日（周日） 23:59

提交内容为 Lab5_学号_姓名.zip 和 Lab5_学号_姓名.pdf，上传至 eLearning，压缩包中应包含所有代码和运行 purchase.txt 和 sell.txt 后的结果，保存在 log.txt 中。pdf 文件为实验文档，实验文档中需要回答第四部分的问题，同时说明代码的实现思路。

提交的压缩包文件目录结构如下：

```
Lab5_学号_姓名.zip:
  AnyElse.cpp
  AnyElse.h
  FamilyMart.h
  log.txt
```

AnyElse 表示其他任意文件名。

评分

评分项目	分值
正确运行提供的 purchase.txt 和 sell.txt	10
正确运行前三天的测试	10
正确运行前十天的测试	10
正确运行三个月的测试	15
正确运行一年的测试	15
代码风格、代码注释	10
设计思路	10
回答问题	20
附加分	10

本次 Lab 满分 100 分，由于难度较大，基础分拿到 100 分并不容易，因此在基础分达到 60 分且未滿 100 分的情况下，可以用附加分补足。

逾期提交按一天 10% 的比例扣除相应分数，例如本次 Lab 得分为 95 分，逾期提交两天，最终得分为 $95 * 80\% = 76$ 分。

不按要求命名文件，扣除 10% 的分数。

严禁抄袭其他同学代码，或学长学姐代码，或网上代码，抄袭 0 分！

特别提醒：为避免内卷，本次 Lab 文档有长度限制，转换成的 pdf 格式文档不得超过 6 页！超出部分均视为不存在，不予阅读或评分！请同学们务必注意突出自己的文档重点，在最短的篇幅内将所需内容阐述清楚。不得通过把字体改得很小（几乎看不见）来强行满足长度限制，比如使用 word 中的「八号」大小等。

提示

前十天的测试数据没有异常问题，可以参考给出的两个样例文件。但如果想要通过更多测试用例，需要对一些可能的问题多加注意，全面考虑。越往后的测试用例，对程序的鲁棒性要求越高。在考虑鲁棒性时，可以比照真实的超市系统，具体的处理方式也可以参照现实生活。凡是正常的真实的超市系统中可能发生的问题，测试中均有可能发生；真实世界不会发生的，测试中也不会发生。例如，可能会有顾客来询问有没有某商品（有可能购买没有库存的商品），但是不会有商品的标价精确到小数点后三位，也不会有一个商品折扣不在 $[0, 1]$ 的范围内，更不会有顾客来买一个 `nullptr`。

由于是通过代码执行和文件比较进行自动化测试，所以请务必注意文件路径，头文件导入等问题。

其他任何关于需求的问题都可以在微信群中提问，技术上的问题可以私聊助教。