

## Lab5 设计文档

回答问题：

重载和默认参数之间的区别？

相同点：

1.函数的重载与函数的默认参数都可以实现函数对于一类问题的处理，因此两者不可同时对同一函数使用，这样的话系统执行时会报错。

不同点：

1.函数的重载可以使得同名函数根据不同形参个数，不同形参顺序，不同形参类型实现不同功能，而函数的默认参数仅可以使函数根据不同形参个数实现不同功能；

2.函数的重载是在编写多个重名函数，而函数的默认参数是在一个函数内实现的。

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？

其他。先读取整行，再截取字符串中折扣部分，如果此处未截取到则将

discount 赋值为 1（即不打折），否则通过 atof 函数直接变成 double 型变量。

代码思路：

按程序运行顺序：

```
class commodity{//商品类
public:
    string name;//商品名字

    double price;//商品价格
```

```
int life;//商品保质期

string product_date;//商品生产日期

int lifespan;//从第一年开始计算的保质期天数
};

commodity *good=new commodity[100000];//商品库

commodity example;//商品类模版——用于覆盖删除已经售出的商品
```

1. 建立商品库，然后读入当日入库商品，并判断是否过期。

过期判断：通过换算从第一年开始至今的天数来判断。

E.g. 7 天保质期、生产于 2022/05/02 的商品总天数为

$(2022 \times 365 + 31 + 28 + 31 + 30 + 2) + 7 = 738159$ （不考虑闰年）如此只要让每次读入文件时当前天数 today 加 1，如果商品数值小于当前天数则过期，在商品库中删除此商品。

2. 依次读入顾客购买商品并在商品库中查找出最邻近过期的此类商品卖给顾客。

3. 返回商品乘折扣的售价并删除此商品。

性能优化：

不遍历所有商品，就找到最早过期的某个商品。

在读入 `purchase.txt` 文件后，对商品库中商品按换算过的保质期总天数大小进行排序，如此只需从头开始比对商品名，一旦查找出则必然是最临近过期的商品，如此可以不遍历所有商品，就找到最早过期的某个商品，在一般情况下，性能便得到了优化。（特殊情况：如果顾客人数较少，而商品入库数量较多，即使用时间复杂度较低的快排算法，总计算次数可能仍大于直接遍历）

