

Project 1 技术报告

姓名：欧穗新 学号：16340173 班级：教务三班

需求分析：

小喵要求的仓库管理系统的功能如下：

1. 查看仓库库存列表功能
2. 查询功能：可以随时查看当前仓库的库存中的某些商品的一系列信息，包括商品名，剩余量
3. 进货功能：（以下两个功能是根据输入自动选择的）
 - （1）在仓库进货时，如果仓库中没有此商品，则为仓库增添新的商品项目
 - （2）在仓库进货时，如果仓库中已有此商品，则增加此商品的库存量
4. 出货功能：
 - （1）在仓库出货时，减少对应商品的库存量
 - （2）在仓库出货时，如果这是货物是此商品的最后一批货（库存量为0），则删除仓库中此商品项目
5. 退出程序时存储仓库信息
6. 记录仓库管理日志

实际功能：

我的仓库管理系统的功能如下（分为管理员和普通用户）：

****其中 2,3,4,5 功能都可以一次性输入几个商品，而不是每次都只能输入一个商品信息***

一、管理员可以执行的功能：

0. 退出程序
1. 查看仓库库存列表功能
2. 查询功能：可以随时查看当前仓库的库存中的某些商品的一系列信息，包括商品名，剩余量，售价（管理员还可以查看成本价）
3. 进货功能：（以下两个功能是根据输入自动选择的）
 - （1）在仓库进货时，如果仓库中没有此商品，则为仓库增添新的商品项目
 - （2）在仓库进货时，如果仓库中已有此商品，则增加此商品的库存量
4. 出货功能：
 - （1）在仓库出货时，减少对应商品的库存量
 - （2）在仓库出货时，如果这是货物是此商品的最后一批货（库存量为0），则删除仓库中此商品项目
5. 更改产品价格功能

6. 管理员可以更改获取管理权限的密码
7. 管理员可以查看所有用户的操作历史记录

二、普通用户可以执行的功能：

0. 退出程序
1. 查看仓库库存列表
2. 查询功能: 可以随时查看当前仓库的库存中的某些商品的一系列信息, 包括商品名, 剩余量, 售价

三、仓库资料库功能（有一个固有文件 **txt1**，存储管理密码和用户数目；一个固有文件存储仓库信息 **txt2**）：

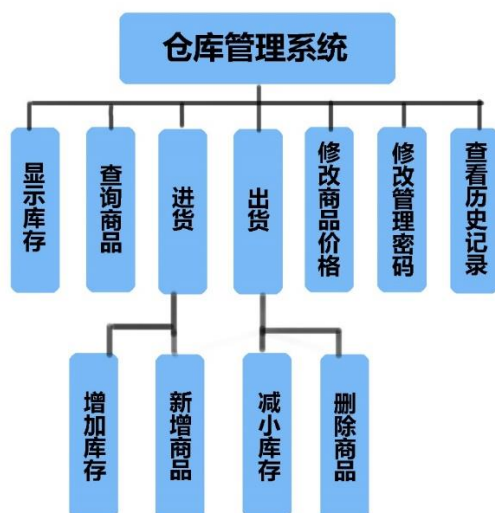
1. 退出程序时存储仓库信息 （每次运行程序时，将 **txt2** 中的仓库信息写入程序，同时将 **txt1** 里面的密码；程序结束前，把仓库信息保存到 **txt2** 里面，同时也保存密码信息进入 **txt1**）
2. 记录仓库管理日志（每个注册的用户都会在注册时产生一个记录其操作历史的文件，文件名就是 ID 号，由 **txt1** 里面的用户总数产生，每次注册 “用户总数” ++，产生该文件。）

四、输入异常处理功能

每次用户输入都会检测用户输入是否正确（一个 **check（）** 函数）

实现思路:

功能结构图：•



本仓库管理系统主要分为出货、进货、查询、列表库存、修改商品价格、修改管理密码、查看历史记录。七大模块，分别对出货、进货、查询、修改商品价格、修改管理密码、查看历史记录等的操作进行管理。

进货模块中又细分为增加库存和新增商品子功能，当进货时，若此商品在仓库中没有库存，则在仓库库存条目中新增此商品项目，若已有此类商品，则根据进货量增加对应的库存量。出货模块中又细分为减少库存和删除商品子功能，当出货时，减少对应商品的数目，注意到

当库存不足时，出货失败，且若出货成功并且库存为 0 时，删除仓库目录中此商品项目。

具体的实现方法是：

设计一个链表，通过遍历链表打印商品信息来 实现显示商品库存的函数；

通过遍历链表，比较用户输入的 **name** 查找商品来 实现查找商品的函数；

然后在进货时，通过查找函数判断库存中是否有该商品，从而决定执行新增商品/增加库存；

同样在出货时，通过查找函数判断库存中是否有该商品，如果有，再进一步判断商品的数目与用户想要的数目，从而选择 **delete** 商品/**decrease** 商品；

修改价格时 通过查找函数得到指向该商品的指针，从而更改价格；

修改管理密码则是直接更改全局变量 **Password**，最后把改完的结果输入 **txt1**；

查看历史记录是根据用户（管理员）输入的 **ID** 打开相应的用户文件，把文件内容写到终端；

通过以上方法，最终实现了：七个主要功能（对普通用户而言只有两个功能，即 1,2 功能）、一个简单加密管理系统、生成文件并存储用户操作记录、存档仓库信息。

数据设计：

```
struct Goods {  
    string name;  
    int count = 0;  
    double prime_costs = 0;  
    double selling_price = 0;  
    Goods *next = NULL;  
}
```

name 表示名称，**count** 表示数量，**prime_cost** 表示进价，**selling_cost** 表示售价，**count** 表示货物数量

其中，管理员可以访问所有变量，并赋予所有变量初始值，并且可以在之后更改售价

用户只能读取除了进价以外的数据，不能进行写操作

函数设计：

一、关于交互信息的函数：

（1）询问用户是不是管理员并审核，如果是 **manager**，返回真且 **ret** 全局变量置为 1，表示表示 **manager**，不是 **manager** 返回真且 **ret=0**，否则输入错误，返回假，要求重新输入

bool is_manager(void);

（2）打印该仓库系统的功能：

int print_function();

(3) 根据用户所选数字确定对应的用户想要的功能, 如果用户选择退出程序, 返回 0; 否则正常执行, 返回 1

```
int choose_function();
```

二、基本函数声明

(1) 进货, 对应进货模块, 表示当前进货一批数量为 count 的 name 商品。

```
void add_goods(string name, int count, double prime_costs, double selling_price);
```

它有两个子函数:

(1.1) 更新库存信息, 对应增加库存子功能, 对 name 商品新增 count 数量

```
void increase_count (string name, int count);
```

(1.2) 更新库存列表, 对应新增商品子功能, 新增 name 商品且初始数量为 count

```
void add_to_list(string name, int count, double prime_costs, double selling_price);
```

(2) 出货, 对应出货模块, 表示当前出货一批数量为 count 的 name 商品, 如果该货物不存在, 返回 0, 否则, 返回 1, 表示成功取出。

```
int delete_goods(string name, int count);
```

它有两个子函数:

(2.1) 更新库存信息, 对应减少库存子功能, 对 name 商品减少 count 数量

```
bool decrease_count(string name, int count);
```

(2.2) 更新库存列表, 对应删除商品子功能, 删除商品列表中 name 商品,

```
bool delete_from_list(string name);
```

(3) 显示当前库存列表, 包括商品名及其库存量, 成本价, 售价; 普通成员不可以看到成本价

```
void show_goods();
```

(4) 查看仓库中的 name 商品的一系列参数, 普通成员不可以看到成本价, 函数返回的是指向查找得到的商品的指针; 如果仓库没有这个商品, 返回一个默认构造的 goods 变量 (这个变量的 count 是 0, 以此判断该商品不存在)

```
Goods* find_goods(string name);
```

三、新增功能函数:

新增的功能, 总共有管理者函数和用户函数两部分:

(1) 管理者函数:

(1.1) 首先应该包括所有的 basicfunction 里面的函数:

即 add_goods, increase_count, add_to_list;

和 delete_goods, decrease_count, delete_from_list

以及 show_goods, find_goods

然后是新增的函数:

(1.2) 修改仓库访问密码:

```
void change_password(string password);
```

(1.3) 修改售价:根据商品名称进行修改:(如果存在这个商品, 返回 1 表示修改成功, 否则返回 0 表示修改)

```
int change_price(string name, double price);
```

(1.4) 管理员查看所有用户操作(包括普通用户和管理员)如果存在该用户 id, 返回 1 表示访问成功, 否则返回 0 表示访问失败

```
int open_file(string id);
```

(2) 用户函数:

```
包括: void show_goods(), Goods* find_goods(string name);
```

(注意这个功能不同于管理者的查询函数, 不可以看到进货价格)

四、存档所有信息函数, 包括密码信息, 仓库信息, 用户操作信息等等

(1) 每次运行程序将之前保存的仓库信息, 密码信息, 用户数目写入程序

```
void write_back(void);
```

(2) 这个函数用来保存关闭程序时仓库的存储情况, 以免退出程序后数据丢失而对应的文件“txt2”, 就是专门用来保存仓库信息的.

```
void Place_on_file(void);
```

(3) 这个函数可以向存储用户信息的文件写入时间信息, 从而记录用户进行某一个操作的时间, 返回操作具体时间

```
string get_time(void);
```

```
//本来应该有一个专门根据用户的选择, 向文件输出, 记录用户操作的函数
```

```
//但是用户每个不同的操作, 有不同的输入, 而每个操作用户只输出一次, 所以不适合把‘写向屏幕’和‘写向文件’的操作分成两个函数
```

```
//如果非要这么做, 要么(1)要么需要传递很多变量, 多分很多情况, 要么(2)需要定义很多全局变量,
```

```
//所以干脆统一写在 choose_function()里面了
```

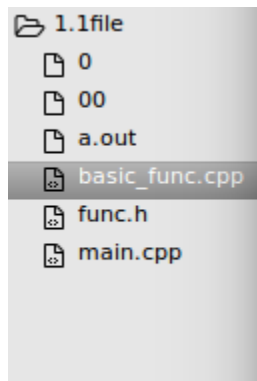
```
//总之,, 水平有限,,,,, 时间精力有限,,,,,
```

五、异常处理函数(2.0 新增)

一个 `void check (int num)`, 用于检测用户输入是否正确。

输入与输出（以下附上图片和说明，从中可以看到输入输出）：

这是一开始的文件状况，0 文件即 Word 说明文档中的 txt1,00 文件对应 txt2:



这是第一次注册后，文件状况（左侧框）和输入输出情况（右侧终端界面）
可以看到（1）文件筐多了“1”文件，这是保存用户 ID 为“1”的用户的操作记录文件
（2）右侧终端中，每一行以“~~”开头的表示程序的输出语句（用于提示用户操作），以“\$\$”开头的表示用户的输入。以此便于区分（图中还进行了第一个功能操作）

```
FOLDERS
└─ c++ files
└─ python files
└─ makefile
└─ 1.0file
└─ 1.1file
    ├── 0
    ├── 00
    ├── 1
    ├── a.out
    ├── basic_func.cpp
    ├── func.h
    └── main.cpp

ousuixin@ubuntu:~/Desktop/1.1file$ g++ main.cpp basic_func.cpp -std=c++11
ousuixin@ubuntu:~/Desktop/1.1file$ ./a.out
~~ Did you have an ID before?
~~ Input y/n to represent yes/no, or input q to quit.
$$ n
~~ Do you want to register ?
~~ Input y/n to represent yes/no, or input q to quit.
$$ y
~~ Register seccessfully!
~~ Now, you are one of the user! Your id is 1.
~~ Are you the managers ?
~~ Input y/n to represent yes/no,
$$ y
~~ Please enter the password:
$$ o
~~ Login seccessfully!
~~ Now you are a manager!
~~ Dear number(1), what do you want to do ?In this storehouse, you can :
#####
##      (0).Exit the program          ##
##      (1).Show The Goods List      ##
##      (2).Search For Specific Goods ##
##      (3).Add Some Goods           ##
##      (4).Take Out Specific Goods  ##
##      (5).Change Price Of Goods    ##
##      (6).Change Password          ##
##      (7).Open A User's FILE       ##
#####
~~ Please enter the numbers to perform the corresponding operations.
$$ 1
#### #####
## GOODS ID  GOODS NAME      GOODS NUMBERS  PRIME COSTS  SELLING PRICE ##
#### #####
~~ There are 0 goods now.

~~ What else do you want to do?
#####
##      (0).Exit the program          ##
##      (1).Show The Goods List      ##
##      (2).Search For Specific Goods ##
##      (3).Add Some Goods           ##
##      (4).Take Out Specific Goods  ##
##      (5).Change Price Of Goods    ##
##      (6).Change Password          ##
##      (7).Open A User's FILE       ##
#####
$$ 0
```

这里执行了第二个操作（由于仓库是空的，拒绝搜索）：

```
1.0file
└─ 1.1file
   └─ 0
      └─ 00
         └─ 1
            └─ a.out
               └─ basic_func.cpp
                  └─ func.h
                     └─ main.cpp

ousuixin@ubuntu:~/Desktop/1.1file$ g++ main.cpp basic_func.cpp -std=c++11
ousuixin@ubuntu:~/Desktop/1.1file$ ./a.out
-- Did you have an ID before?
-- Input y/n to represent yes/no, or input q to quit.
$$ y
-- Please enter your id.
$$ 1
-- Are you the managers ?
-- Input y/n to represent yes/no,
$$ y
-- Please enter the password:
$$ o
-- Login seccessfully!
-- Now you are a manager!
-- Dear number(1), what do you want to do ?In this storehouse, you can :
#####
##      (0).Exit the program      ##
##      (1).Show The Goods List   ##
##      (2).Search For Specific Goods ##
##      (3).Add Some Goods        ##
##      (4).Take Out Specific Goods ##
##      (5).Change Price Of Goods  ##
##      (6).Change Password       ##
##      (7).Open A User's _FILE   ##
#####
-- Please enter the numbers to perform the corresponding operations.
$$ 2
-- How many Goods do you want to find?
$$ 1
-- Please input names of the goods.
$$ mo
-- There is no such goods named mo.
-- What else do you want to do?
#####
##      (0).Exit the program      ##
##      (1).Show The Goods List   ##
##      (2).Search For Specific Goods ##
##      (3).Add Some Goods        ##
##      (4).Take Out Specific Goods ##
##      (5).Change Price Of Goods  ##
##      (6).Change Password       ##
##      (7).Open A User's _FILE   ##
#####
$$ 0
```

执行第三功能，连续添加四种商品：

```
FOLDERS
└─ c++ files
   └─ python files
      └─ makefile
         └─ 1.0file
            └─ 1.1file
               └─ 0
                  └─ 00
                     └─ 1
                        └─ a.out
                           └─ basic_func.cpp
                              └─ func.h
                                 └─ main.cpp

-- Please enter your id.
$$ 1
-- Are you the managers ?
-- Input y/n to represent yes/no,
$$ y
-- Please enter the password:
$$ o
-- Login seccessfully!
-- Now you are a manager!
-- Dear number(1), what do you want to do ?In this storehouse, you can :
#####
##      (0).Exit the program      ##
##      (1).Show The Goods List   ##
##      (2).Search For Specific Goods ##
##      (3).Add Some Goods        ##
##      (4).Take Out Specific Goods ##
##      (5).Change Price Of Goods  ##
##      (6).Change Password       ##
##      (7).Open A User's _FILE   ##
#####
-- Please enter the numbers to perform the corresponding operations.
$$ 3
-- How many Goods do you want to add?
$$ 4
-- Please input name numbers, prime_costs, and selling_price of the goods.
$$ mouse 4 5.9 10
-- Add goods seccessfully!
-- Please input name numbers, prime_costs, and selling_price of the goods.
$$ screen 6 233 300
-- Add goods seccessfully!
-- Please input name numbers, prime_costs, and selling_price of the goods.
$$ CD 10 300 500
-- Add goods seccessfully!
-- Please input name numbers, prime_costs, and selling_price of the goods.
$$ keyboard 8 23 30
-- Add goods seccessfully!
-- What else do you want to do?
#####
##      (0).Exit the program      ##
##      (1).Show The Goods List   ##
##      (2).Search For Specific Goods ##
##      (3).Add Some Goods        ##
##      (4).Take Out Specific Goods ##
##      (5).Change Price Of Goods  ##
##      (6).Change Password       ##
##      (7).Open A User's _FILE   ##
#####
$$ 0
```

这时候再使用第一个函数，就有商品列表了：

```
c++ files
python files
makefile
1.0file
1.1file
  0
  00
  1
  a.out
  basic_func.cpp
  func.h
  main.cpp

-- Did you have an ID before?
-- Input y/n to represent yes/no, or input q to quit.
$ y
-- Please enter your id.
$ 1
-- Are you the managers ?
-- Input y/n to represent yes/no,
$ y
-- Please enter the password:
$ 0
-- Login seccessfully!
-- Now you are a manager!
-- Dear number(1), what do you want to do ?In this storehouse, you can :
#####
##      (0).Exit the program      ##
##      (1).Show The Goods List   ##
##      (2).Search For Specific Goods ##
##      (3).Add Some Goods        ##
##      (4).Take Out Specific Goods ##
##      (5).Change Price Of Goods ##
##      (6).Change Password       ##
##      (7).Open A User's FILE   ##
#####
-- Please enter the numbers to perform the corresponding operations.
$ 1
#### #####
## GOODS_ID  GOODS_NAME  GOODS_NUMBERS  PRIME_COSTS  SELLING_PRICE  ##
## (1)      CD          10             300          500           ##
## (2)      keyboard    8              23           30            ##
## (3)      mouse       4              5.9          10            ##
## (4)      screen      6              233          300           ##
#### #####
-- There are 4 goods now.

-- What else do you want to do?
#####
##      (0).Exit the program      ##
##      (1).Show The Goods List   ##
##      (2).Search For Specific Goods ##
##      (3).Add Some Goods        ##
##      (4).Take Out Specific Goods ##
##      (5).Change Price Of Goods ##
##      (6).Change Password       ##
##      (7).Open A User's FILE   ##
#####
$ 0
```

这时候执行第二个功能，就可以进行（1）查找，并且（2）如果没有该商品，会指出没有商品：

```
c++ files
python files
makefile
1.0file
1.1file
  0
  00
  1
  a.out
  basic_func.cpp
  func.h
  main.cpp

-- Your input is error! Please enter the right one!
$ y
-- Please enter your id.
$ 1
-- Are you the managers ?
-- Input y/n to represent yes/no,
$ y
-- Please enter the password:
$ 0
-- Login seccessfully!
-- Now you are a manager!
-- Dear number(1), what do you want to do ?In this storehouse, you can :
#####
##      (0).Exit the program      ##
##      (1).Show The Goods List   ##
##      (2).Search For Specific Goods ##
##      (3).Add Some Goods        ##
##      (4).Take Out Specific Goods ##
##      (5).Change Price Of Goods ##
##      (6).Change Password       ##
##      (7).Open A User's FILE   ##
#####
-- Please enter the numbers to perform the corresponding operations.
$ 2
-- How many Goods do you want to find?
$ mo
-- Your input is error, please enter the right one!
$ 2
-- Please input names of the goods.
$ mouse
-- There are 4 mouse left in the storehouse.
-- The selling price of it is 10.
-- The prinme price of it is 5.9.
$ mo
-- There is no such goods named mo.
-- What else do you want to do?
#####
##      (0).Exit the program      ##
##      (1).Show The Goods List   ##
##      (2).Search For Specific Goods ##
##      (3).Add Some Goods        ##
##      (4).Take Out Specific Goods ##
##      (5).Change Price Of Goods ##
##      (6).Change Password       ##
##      (7).Open A User's FILE   ##
#####
$ 0
```

此时再次进行 add，可以看到仓库中（1）mouse 增加了 4 件（2）新增了 musicplayer 5 件：


```
makefile
1.0file
1.1file
0
00
1
a.out
basic_func.cpp
func.h
main.cpp

-- Please enter the numbers to perform the corresponding operations.
$$ 3
-- How many Goods do you want to add?
$$ 2
-- Please input name numbers, prime_costs, and selling_price of the goods.
$$ mouse 4 5.9 10
-- Add goods seccessfully!
-- Please input name numbers, prime_costs, and selling_price of the goods.
$$ musicplayer 5 1200 2000
-- Add goods seccessfully!
-- What else do you want to do?
#####
## (0).Exit the program ##
## (1).Show The Goods List ##
## (2).Search For Specific Goods ##
## (3).Add Some Goods ##
## (4).Take Out Specific Goods ##
## (5).Change Price Of Goods ##
## (6).Change Password ##
## (7).Open A User's FILE ##
#####
$$ 1
#### ##
## GOODS_ID GOODS_NAME GOODS_NUMBERS PRIME_COSTS SELLING_PRICE ##
## (1) CD 10 300 500 ##
## (2) keyboard 8 23 30 ##
## (3) mouse 8 5.9 10 ##
## (4) musicplayer 5 1200 2000 ##
## (5) screen 6 233 300 ##
#### ##
-- There are 5 goods now.
-- What else do you want to do?
#####
## (0).Exit the program ##
## (1).Show The Goods List ##
## (2).Search For Specific Goods ##
## (3).Add Some Goods ##
## (4).Take Out Specific Goods ##
## (5).Change Price Of Goods ##
## (6).Change Password ##
## (7).Open A User's FILE ##
#####
$
```

执行功能四，可以看到（1）取出 CD 件数为 8，超过库存时，拒绝了操作，（2）取出件数与库存相等时，删除了该商品，所以之后库存没了 CD：

（还有一张图，关于功能四我搞丢了：（3）可以看到 没有商品时拒绝取出（4）商品数目大于要求数目，该商品数目 decrease 了）

```
c++ files
python files
makefile
1.0file
1.1file
0
00
1
a.out
basic_func.cpp
func.h
main.cpp

#####
## (7).Open A User's FILE ##
#####
$$ 4
-- How many Goods do you want to take out?
$$ 1
-- Please input name and numbers of the goods.
$$ CD 8
-- The number of goods named CD is not enough ,so you can not take them out
-- What else do you want to do?
#####
## (0).Exit the program ##
## (1).Show The Goods List ##
## (2).Search For Specific Goods ##
## (3).Add Some Goods ##
## (4).Take Out Specific Goods ##
## (5).Change Price Of Goods ##
## (6).Change Password ##
## (7).Open A User's FILE ##
#####
$$ 4
-- How many Goods do you want to take out?
$$ 1
-- Please input name and numbers of the goods.
$$ CD 7
-- Take goods seccessfully!
-- What else do you want to do?
#####
## (0).Exit the program ##
## (1).Show The Goods List ##
## (2).Search For Specific Goods ##
## (3).Add Some Goods ##
## (4).Take Out Specific Goods ##
## (5).Change Price Of Goods ##
## (6).Change Password ##
## (7).Open A User's FILE ##
#####
$$ 1
#### ##
## GOODS_ID GOODS_NAME GOODS_NUMBERS PRIME_COSTS SELLING_PRICE ##
## (1) keyboard 8 23 30 ##
## (2) mouse 8 5.9 10 ##
## (3) musicplayer 5 1200 2000 ##
## (4) screen 3 233 300 ##
#### ##
-- There are 4 goods now.
-- What else do you want to do?
```

我们执行了 5，连续更改两个商品价格，看到了商品 mouse，screen 价格都改变成输入的数字，再次执行 5，看到：如果没有这个商品，则拒绝更改商品价格：


```
1.1file
├── 0
├── 00
├── 1
├── a.out
├── basic_func.cpp
├── func.h
└── main.cpp

~~ What else do you want to do?
#####
##          (0).Exit the program          ##
##          (1).Show The Goods List       ##
##          (2).Search For Specific Goods ##
##          (3).Add Some Goods            ##
##          (4).Take Out Specific Goods   ##
##          (5).Change Price Of Goods     ##
##          (6).Change Password           ##
##          (7).Open A User's _FILE      ##
#####
$$ 6
~~ Please enter the password you want.
$$ ousuixin_123456
~~ Set password seccessfully!
~~ What else do you want to do?
#####
##          (0).Exit the program          ##
##          (1).Show The Goods List       ##
##          (2).Search For Specific Goods ##
##          (3).Add Some Goods            ##
##          (4).Take Out Specific Goods   ##
##          (5).Change Price Of Goods     ##
##          (6).Change Password           ##
##          (7).Open A User's _FILE      ##
#####
```

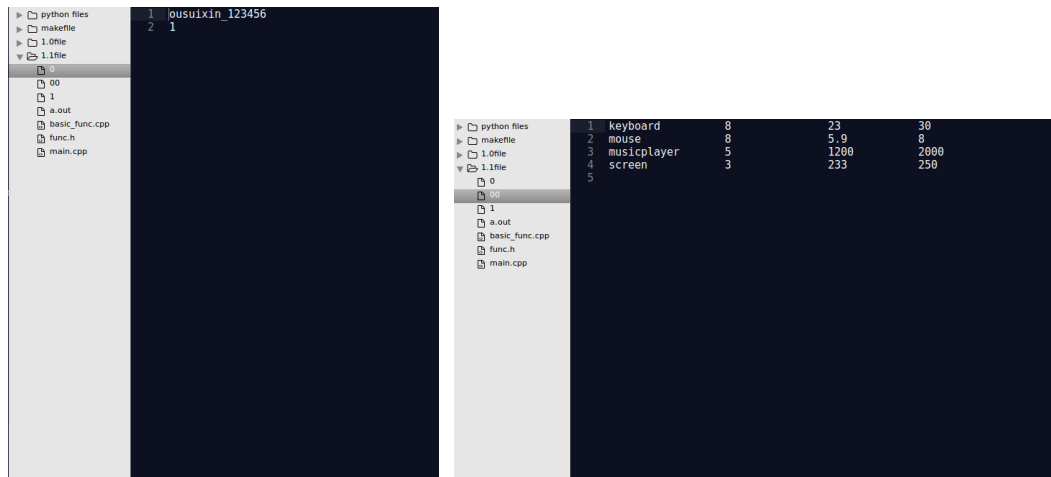
执行了功能 7，通过输入用户自己的 ID，看到了自己的操作记录：

```
makefile
├── 1.0file
└── 1.1file
    ├── 0
    ├── 00
    ├── 1
    ├── a.out
    ├── basic_func.cpp
    ├── func.h
    └── main.cpp

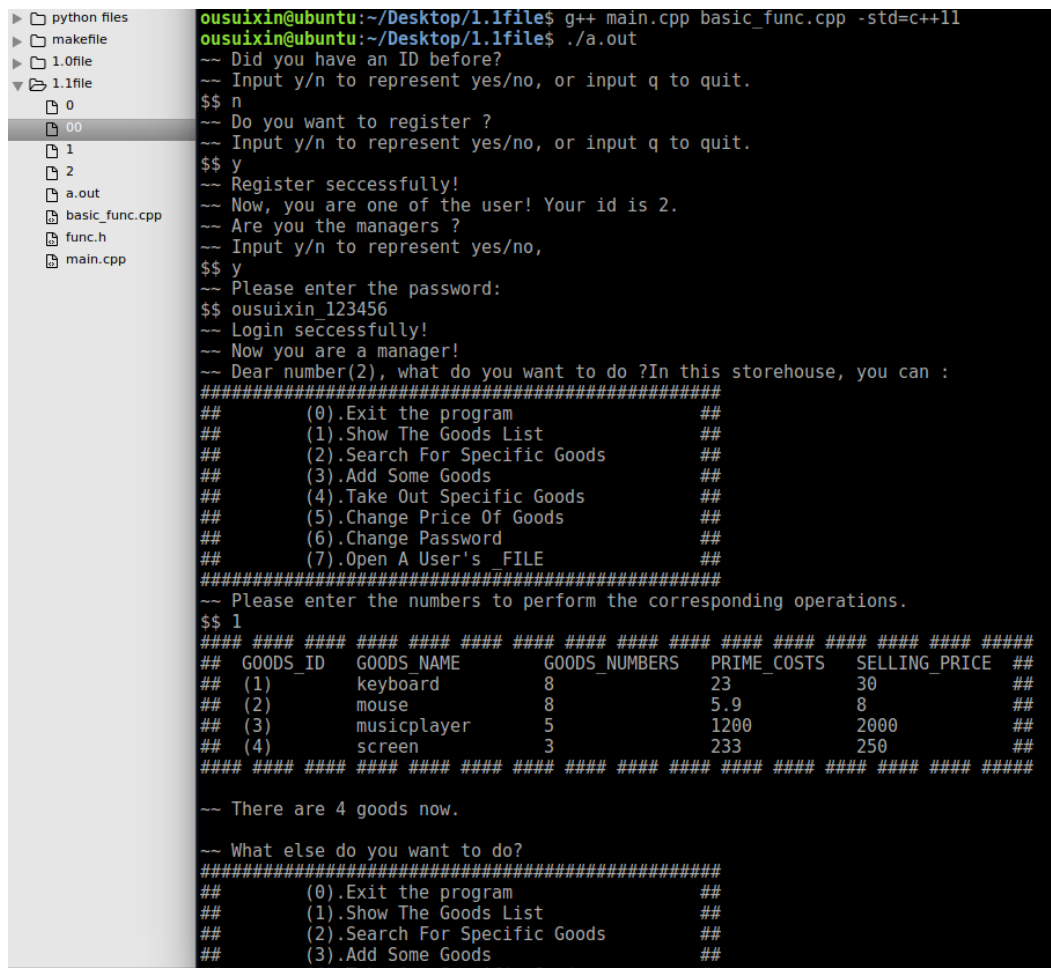
$$ 7
~~ Please enter the id of the user whose operations history you want to see.
$$ 1
~~ 2017/03/19 00:37:48 Sunday
Ask for list all goods.
~~ 2017/03/19 00:41:42 Sunday
Ask for find goods named mo, but there is no such goods.
~~ 2017/03/19 00:43:49 Sunday
Ask for find goods named mo, but there is no such goods.
~~ 2017/03/19 00:45:19 Sunday
Add 4 goods named mouse, the prime_costs of one is 5.9, the prime_costs of one is set to 10.
~~ 2017/03/19 00:45:32 Sunday
Add 6 goods named screen, the prime_costs of one is 233, the prime_costs of one is set to 300.
~~ 2017/03/19 00:45:43 Sunday
Add 10 goods named CD, the prime_costs of one is 300, the prime_costs of one is set to 500.
~~ 2017/03/19 00:45:59 Sunday
Add 8 goods named keyboard, the prime_costs of one is 23, the prime_costs of one is set to 30.
~~ 2017/03/19 00:47:03 Sunday
Ask for list all goods.
~~ 2017/03/19 00:48:25 Sunday
Ask for find goods named mouse, there are 4 mouse left in the storehouse.
The selling price of it is 10.
~~ 2017/03/19 00:48:25 Sunday
The prinme price of it is 5.9.
~~ 2017/03/19 00:48:28 Sunday
Ask for find goods named mo, but there is no such goods.
~~ 2017/03/19 00:50:52 Sunday
Add 4 goods named mouse, the prime_costs of one is 5.9, the prime_costs of one is set to 10.
~~ 2017/03/19 00:51:14 Sunday
Add 5 goods named musicplayer, the prime_costs of one is 1200, the prime_costs of one is set to 2000.
~~ 2017/03/19 00:51:21 Sunday
Ask for list all goods.
~~ 2017/03/19 00:53:54 Sunday
Take out 3 goods named CD.
~~ 2017/03/19 00:54:16 Sunday
Take out 3 goods named screen.
~~ 2017/03/19 00:54:32 Sunday
Ask for take out the goods named 1, but there is no such goods.
~~ 2017/03/19 00:54:40 Sunday
Ask for list all goods.
~~ 2017/03/19 00:56:49 Sunday
Ask for take out 8 goods named CD, but there is no enough such goods.
~~ 2017/03/19 00:57:04 Sunday
Take out 7 goods named CD.
~~ 2017/03/19 00:57:08 Sunday
Ask for list all goods.
```

退出程序之后看到，（1）原来的 0 文件保存的初始密码“0”；
（2）变成了“ousuixin_123456”，用户数由初始的 0，变成了 1；

(3) 仓库库存由空变成了四件:

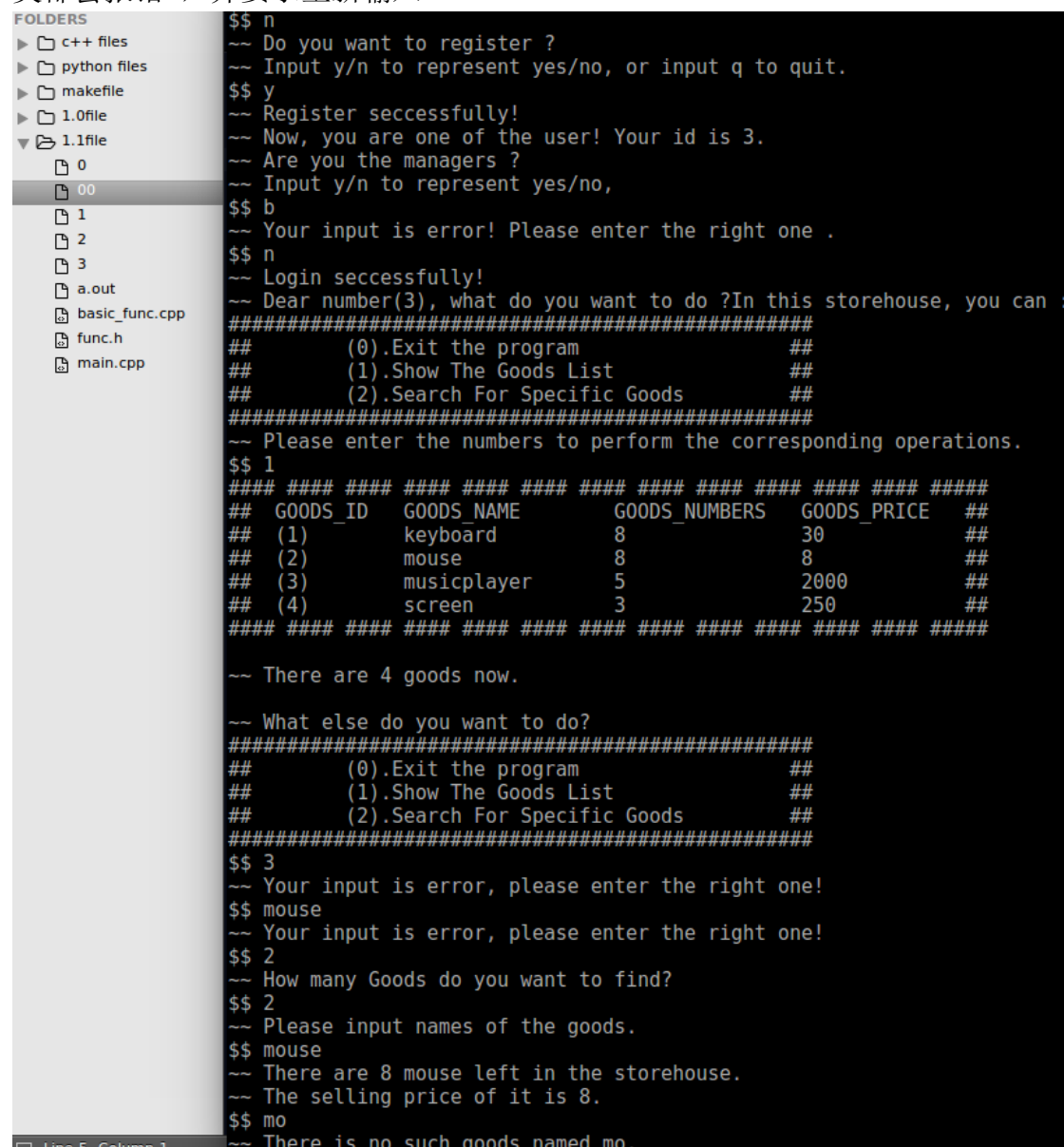


我们又注册了一个账户，发现文件夹里面又多了“2”文件即保存新用户历史记录的文件：



我们又注册了新用户，(1) 文件夹又多了一个名为“3”的文件，保存新用户操作记录，但是(2)这个用户不是管理员，所以只有两个功能，两个功能均与管理员功能相似，但是看

不到成本价。(3) 同时可以看到，要求输入数字却非法输入字符串，其他数字之类都会报错，并要求重新输入



```
FOLDERS
└─ c++ files
└─ python files
└─ makefile
└─ 1.0file
└─ 1.1file
  └─ 0
  └─ 00
  └─ 1
  └─ 2
  └─ 3
  └─ a.out
  └─ basic_func.cpp
  └─ func.h
  └─ main.cpp

$$$ n
~~ Do you want to register ?
~~ Input y/n to represent yes/no, or input q to quit.
$$$ y
~~ Register seccessfully!
~~ Now, you are one of the user! Your id is 3.
~~ Are you the managers ?
~~ Input y/n to represent yes/no,
$$$ b
~~ Your input is error! Please enter the right one .
$$$ n
~~ Login seccessfully!
~~ Dear number(3), what do you want to do ?In this storehouse, you can :
#####
##          (0).Exit the program          ##
##          (1).Show The Goods List       ##
##          (2).Search For Specific Goods ##
#####
~~ Please enter the numbers to perform the corresponding operations.
$$$ 1
#### #### #### #### #### #### #### #### #### ####
## GOODS_ID  GOODS_NAME      GOODS_NUMBERS  GOODS_PRICE  ##
## (1)       keyboard        8          30      ##
## (2)       mouse           8          8       ##
## (3)       musicplayer      5          2000   ##
## (4)       screen          3          250    ##
#### #### #### #### #### #### #### #### #### ####
~~ There are 4 goods now.

~~ What else do you want to do?
#####
##          (0).Exit the program          ##
##          (1).Show The Goods List       ##
##          (2).Search For Specific Goods ##
#####
$$$ 3
~~ Your input is error, please enter the right one!
$$$ mouse
~~ Your input is error, please enter the right one!
$$$ 2
~~ How many Goods do you want to find?
$$$ 2
~~ Please input names of the goods.
$$$ mouse
~~ There are 8 mouse left in the storehouse.
~~ The selling price of it is 8.
$$$ mo
~~ There is no such goods named mo.
```

总结:

大概功能，实现方法，输入输出...上面已经列出

(2.0 版本修复若干 bug: 没有内存泄漏，用 valgrind 试过了；异常处理也做了。)

最后讲一下缺点吧

- (1) 本来还想实现每个用户都有一个密码（包括管理员，普通用户，但是管理员还有管理权限密码，即管理员有两个密码），另外有一个管理员才知道的管理权限密码，并且一旦管理员注册过后，系统会知道这个 ID 的用户是管理员，不要再输入管理权限密码就可以进行管理员操作但是精力，时有限，无法完成了

(2) (还有，自己的代码重复度很高，查找算法也不好，还可以优化，但是时间，，，精力，，，不足，就没有继续优化某些代码了)

最后辛苦老师，ta 们看我的破代码 orz，(又长又 low，，，没时间改进了 orz)
o(∩_∩)o 谢谢你们!