# 实验日志

221300054 李子博

**日期：12月1日**

**1.实验内容：需求分析（管理员部分）**

本次作业需要设计一个冬奥纪念品买卖交易平台。程序主要分为两个部分：管理员部分和用户部分。

在main函数需要对一些变量进行初始化，并且展示主菜单。此外还需要记录用户输入的操作，并通过对它的判别来调用相应函数执行操作。

以下内容将对管理员部分需要实现的需求进行详细分析。

程序中管理员只有一位，因此不需要注册功能，但是管理员需要实现登录功能。这部分需要对用户输入的用户名和密码进行判断，如果不正确则登陆失败，并且返回到程序的主菜单。如果正确则进入登录状态，执行下一个阶段的程序。

管理员部分需要单独设计一个大函数，用以将各个分函数集成起来。首先我们需要展示菜单，以便提示用户需要输入哪些数字才能执行相应的操作。此外也需要引入相关变量，记录用户的操作并调用相应函数进行执行。管理员共有七个功能，分别是查看商品、搜索商品、下架商品、查看订单、查看用户、封禁用户和注销。

对于查看所有商品的功能，需要从商品文件中读入所有数据并在程序中完成输出。如果文件中尚且没有商品，则输出“尚且没有商品！”。对于搜索商品功能，需要先读入所有商品信息，在根据用户输入的数据进行循环匹配，并将所有的匹配结果（可能不止一个）输出。对于下架商品功能，则需要先读入数据，然后根据管理员输入的ID进行循环匹配，直到找到匹配结果。对于被匹配的商品，将其状态修改为“已下架”后项文件输出所有数据。若未匹配到商品ID，则输出相关提示后返回管理员主界面。

对于查看所有订单的操作，则需要先从订单文件中读入所有数据，然后在程序中完成输出。对于查看所有用户的操作，则需要先从用户文件中读入所有信息，然后再程序中完成输出。上述两个功能都需要在文件中没有数据时输出相应提示。

对于封禁用户的操作，则需要先从用户文件中读入所有用户的信息，根据管理员输入的用户ID进行循环匹配，直到匹配到相应的用户，则将其状态修改为“封禁”，然后向文件中输出所有信息。若未匹配到用户，则输出相应提示并返回管理员主界面。

最后是注销功能。当管理员输入注销操作时，则需要输出注销提示，然后返回到程序主菜单界面。

**日期：12月2日**

**1.实验内容：需求分析（用户部分）**

以下内容是对用户部分的需求分析。

用户方面功能较多，需要先用一个主体函数将相关功能以函数调用的形式集成到这个主要函数中。

用户部分在宏观层面上分为两个方向：注册和登录。

用户注册是向用户文件中添加新的数据。因此需要先创建用户计数器，从用户文件中读入所有信息，并在最后对用户计数器加一，之后提示用户键入相关信息，将其读入程序中。同时需要由程序根据用户计数器来赋予新用户一个ID名称。在这些完成以后，将新的用户信息连同以前的用户信息一并输出到文件中进行保存。在这个过程中还需要检查用户输入的用户名和密码是否合法，比如用户名是否与其他用户重复，用户密码位数是否正确等等。

用户登录则是对用户文件中已有的数据进行判定和处理。首先需要对用户输入的用户名进行循环匹配，直到匹配到相应的用户数据，然后提示用户输入密码，对匹配到的用户数据中的密码与输入的密码的一致性进行判定，如果一直则登录成功，否则登陆失败，返回用户主菜单。若无法匹配到相应用户数据，则提示用户重新输入用户名称。

在用户登录成功以后，就可以展示用户界面的菜单提示用户选择自己的身份了。这里用户共有卖家买家两种身份，此外还可以选择查看自己的个人信息。

当用户选择买家身份后，需要展示买家功能菜单。买家功能包括：查看商品列表、购买商品、搜索商品、查看历史订单、查看商品详细信息和返回用户主界面。用户的查看商品列表功能需要先从商品文件中读入相关信息，然后筛选其中状态值为“在售”的商品信息，将这些信息通过程序完成输出。用户的购买商品功能，首先需从文件中读入商品和用户的信息，根据用户输入的商品ID进行筛选，若未匹配到可售的商品，则输出相应的提示，请用户重新输入商品ID。若成功匹配，则提示用户输入购买数量（注意数量一定要在商品待售数量之下，并且用户能付得起这些商品的所耗金额），随后在该商品信息中的数量一栏减掉已经售出的数量，在对应用户信息中的余钱一栏中，减去其本次购买花费的钱数。最后将相应信息保存到文件中。最后将相应的信息整合保存到订单文件中。

用户的搜索商品功能与管理员的相应功能基本类似，只需保证他所查看的商品状态为“在售”即可。而用户的查看历史订单功能则也大同小异，先从订单文件中读入全部的订单信息。根据用户本身的用户ID筛选他对应的订单，并输出全部相应的信息。用户的查找商品的详细信息功能实际上是精确匹配的查找机制。在读入文件中的信息以后，需要根据用户输入的商品ID进行循环匹配，直到找到目标商品。之后输出该商品的详细信息。

当用户选择返回用户主界面操作时，即执行函数返回操作，回到上一层级。

对卖家而言，共有六个功能。分别是：发布商品、查看发布商品、修改商品信息、下架商品、查看历史订单和返回用户主界面。

卖家的发布商品功能和用户注册功能相近，区别只在于所需调用的文件不同。发布商品功能需要先识别出当前登录的用户ID，以便于后续保存。同时需要打开商品文件，病毒如相关信息。之后提示用户输入商品的各类信息，并将发布者的相关信息一并保存至该商品的信息中。之后将已读入的信息连同用户新输入的信息一并输出到文件中。

卖家的查看发布商品功能，需要在读入文件中的信息以后，利用当前卖家的ID对读入的商品ID进行检索，输出那些与当前用户ID相符合的商品信息即为当前用户发布的商品信息。修改商品信息功能也需要先匹配查找到相关的商品，然后对其特定的属性（价格或描述）进行修改。最后将修改后的信息输出到文件中实现保存。下架商品也需要先匹配查找到文件中的相关商品，然后将其状态修改为“已下架”，最后输出到文件中保存。

卖家的查看历史订单功能与买家的相关功能几乎一模一样，区别仅仅在查找匹配时需要按照卖家ID进行查找。这里就不再赘述了。

当用户选择返回用户主界面操作时，则执行函数的返回功能，跳出当前层级，回到上一层级。

此外，用户界面还包含个人信息管理功能。这个界面包含四个功能：查看信息、修改信息、充值和返回用户主界面。首先是查看信息功能。此功能只需读入用户文件，匹配并输入当前用户的详细信息即可。修改信息则是在上述操作的基础上，重新输入当前用户的某些信息，覆盖掉以前的信息，并且将其全部输出到文件中。需要注意的是要对新输入的信息的合法性进行检查，如判断新信息与原信息是否重复，以及新输入的用户名是否已被占用。最后是充值功能。在匹配到当前用户以后，充值功能可以修改用户的余钱信息，并将修改后的信息输出到文件中保存。

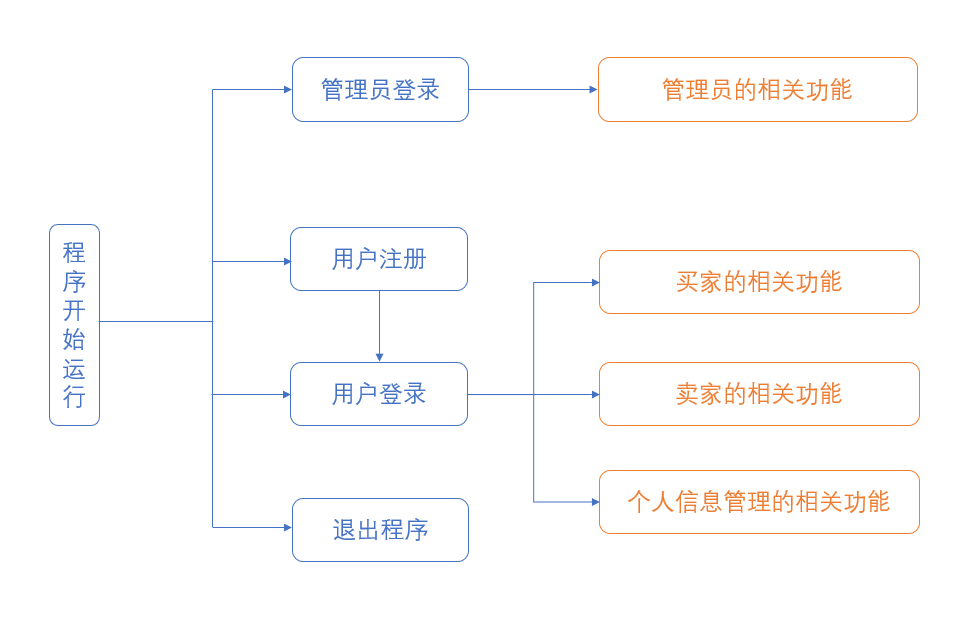
当用户选择返回用户主界面操作时，执行函数的返回功能，跳出当前层级，回到上一层级。

至此，整个程序的所有需求已分析完毕。后续将对本程序进行系统设计。

**日期：12月3日**

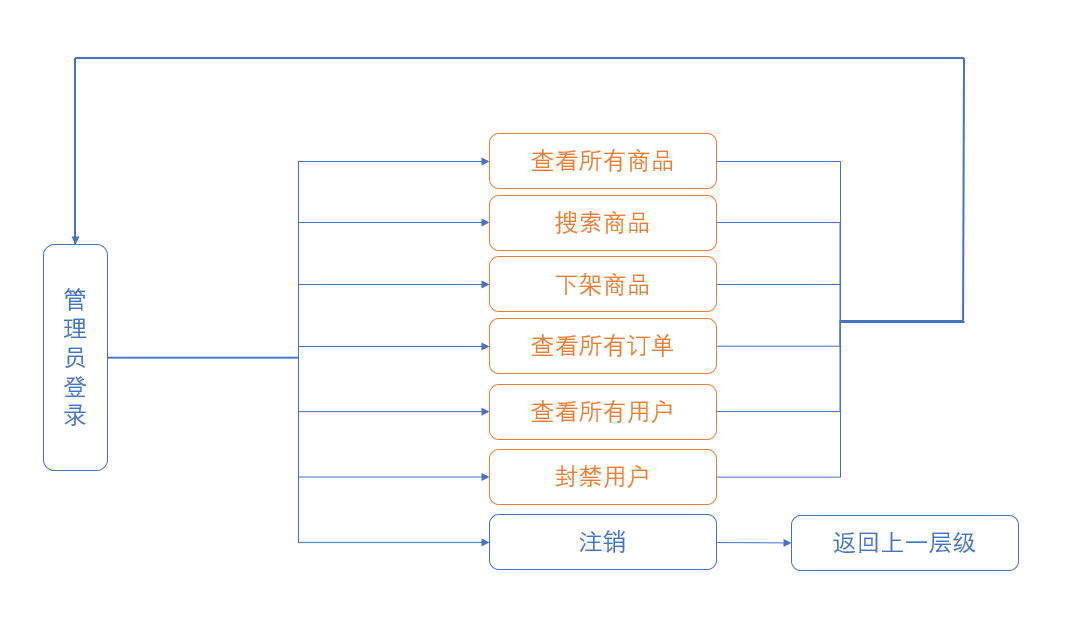
**1.实验内容：系统设计**

以下是对程序整体框架的系统设计思路：



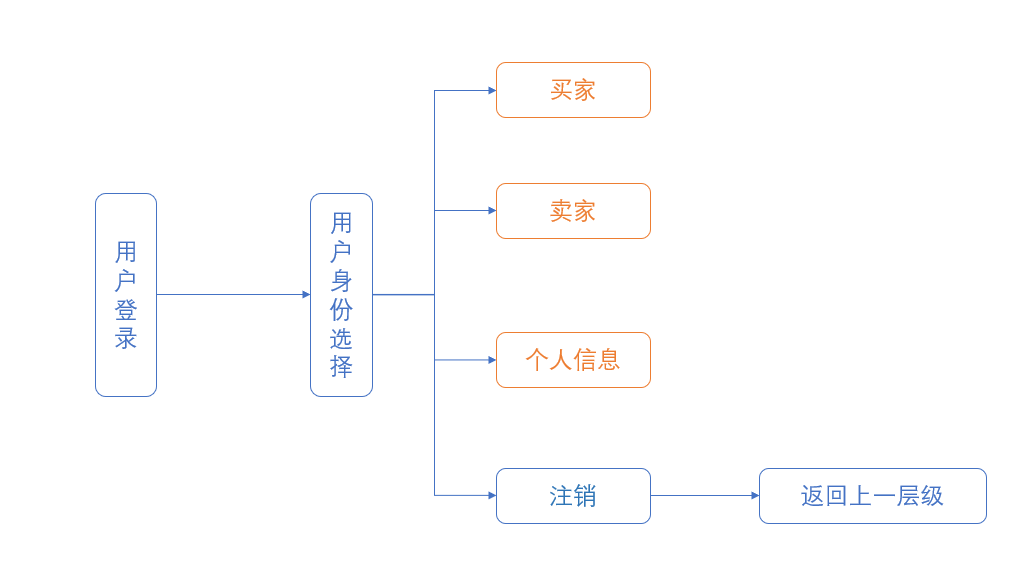
程序主体的四个部分分别在程序开始运行后宫用户进行选择，其中用户注册后随即就可登录，并在登录后相应的使用管理员、买家或买家的相关功能，或者对个人信息进行修改。

以下是管理员相关功能的系统设计思路：



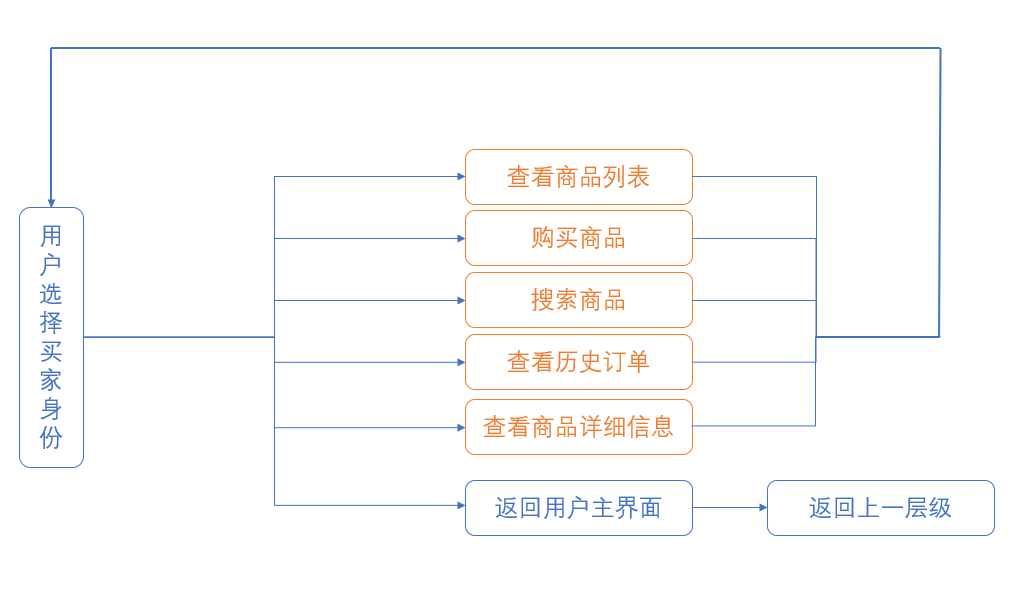
在用户选择管理员身份后，则可以相应的选择管理员的各种功能，在完成一次功能调用后回到管理员功能选择界面，知道用户选择注销，则退出当前层级，回到上一层级。

以下是用户登录的系统设计思路：



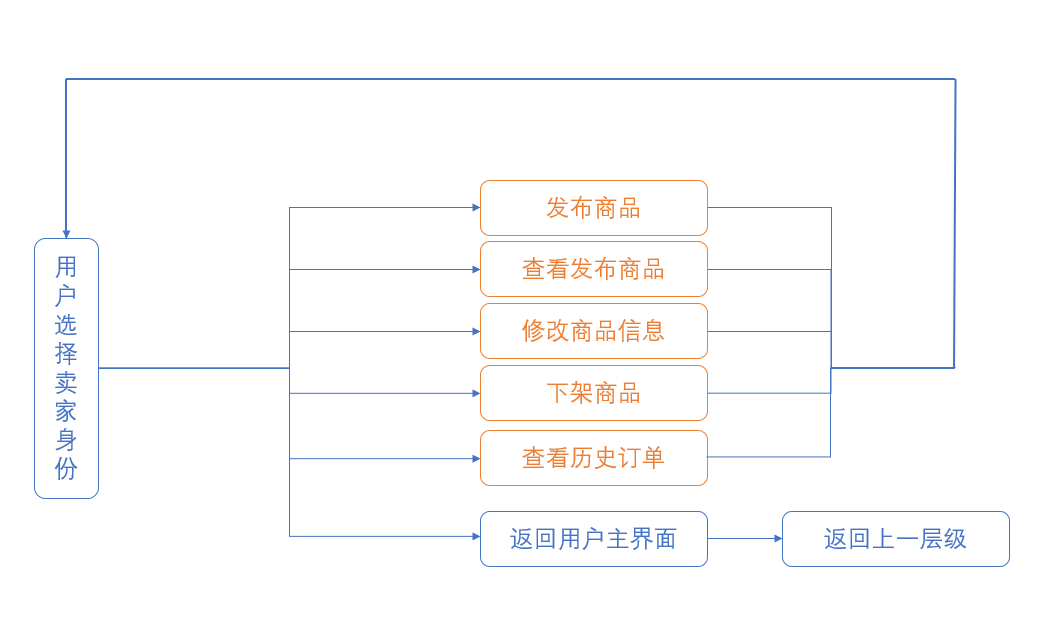
在用户成功登录以后，提示用户对其身份进行选择，并分别对不同身份实现其功能。如果用户选择注销，则直接退出当前层级，返回上一层级。

以下是买家功能的系统设计：



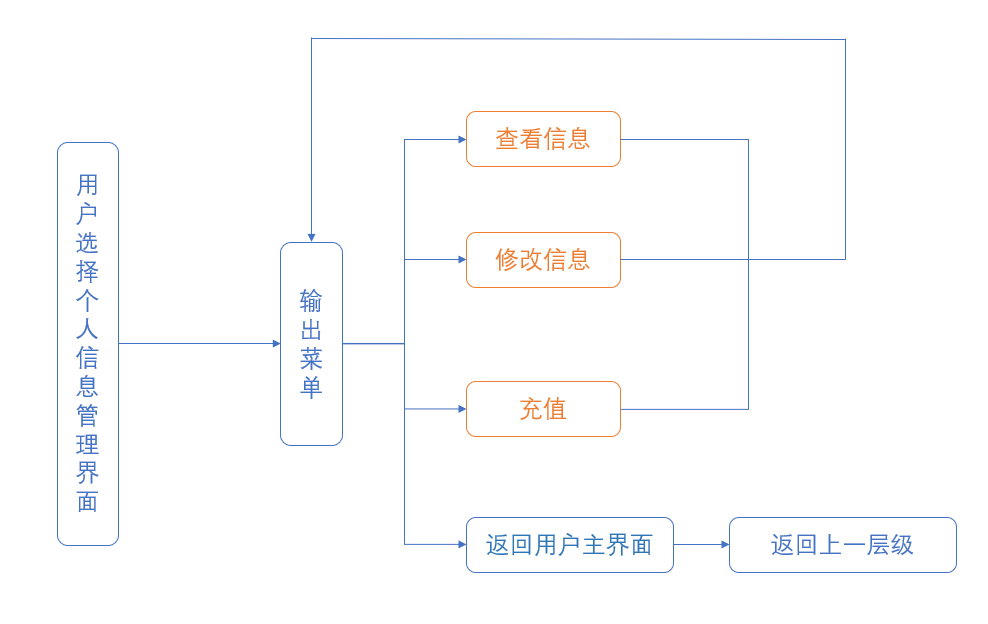
当用户选择买家的身份后，首先输出菜单，提示用户从查看商品列表、购买商品、搜索商品、查看历史订单、查看商品详细信息和返回用户主界面等功能中选择一项来执行。完成每次执行后都返回到输出菜单的步骤，直到用户选择返回用户主界面，则退出当前层级，返回上一层级。

以下是卖家功能的系统设计思路：



当用户选择卖家的身份后，首先输出菜单，提示用户从发布商品、查看发布商品、修改商品信息、下架商品、查看历史订单和返回用户主界面等功能中选择一项来执行。完成每次执行后都返回到输出菜单的步骤，直到用户选择返回用户主界面，则退出当前层级，返回上一层级。

以下是用户个人信息的系统设计思路：

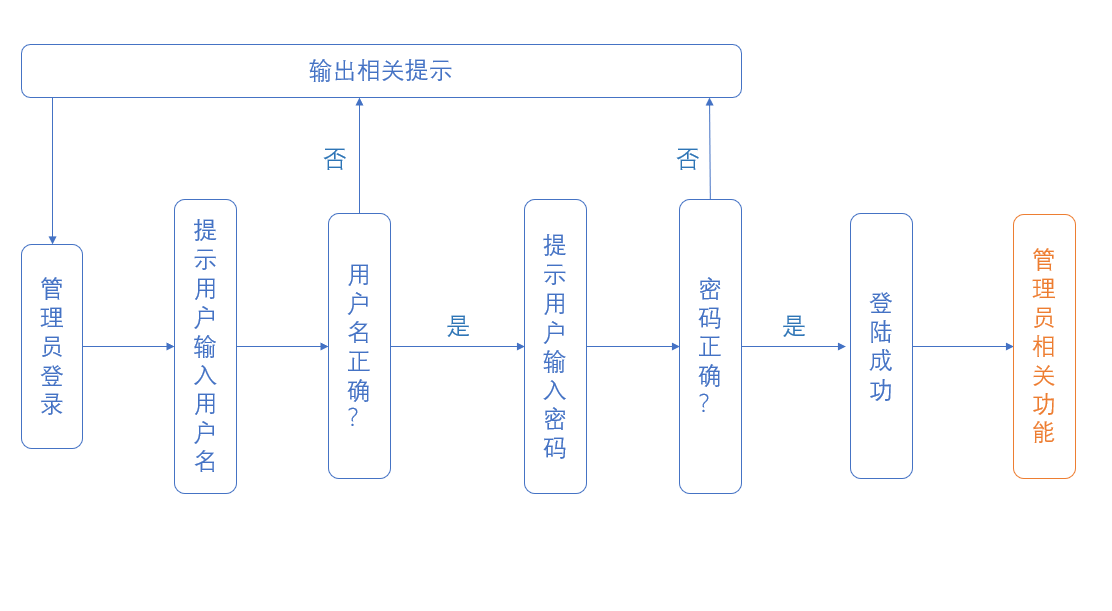


当用户选择了个人信息管理界面时，首先输出菜单，然后提示用户输入相应的选项一致性相应的功能。每次执行完毕后都循环回到输入操作的步骤，直到输入返回用户主界面操作，此时退出当前层级，返回上一层级。

**日期：12月4日**

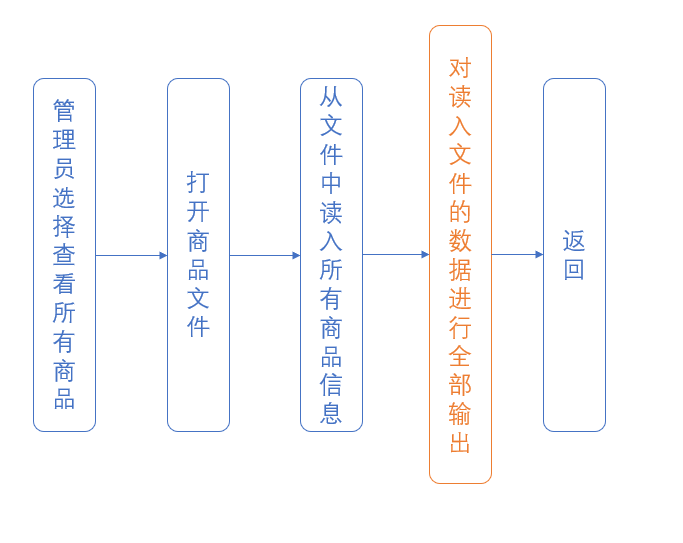
**1.实验内容：详细设计（管理员部分）**

以下是管理员登录相关函数的详细设计思路：



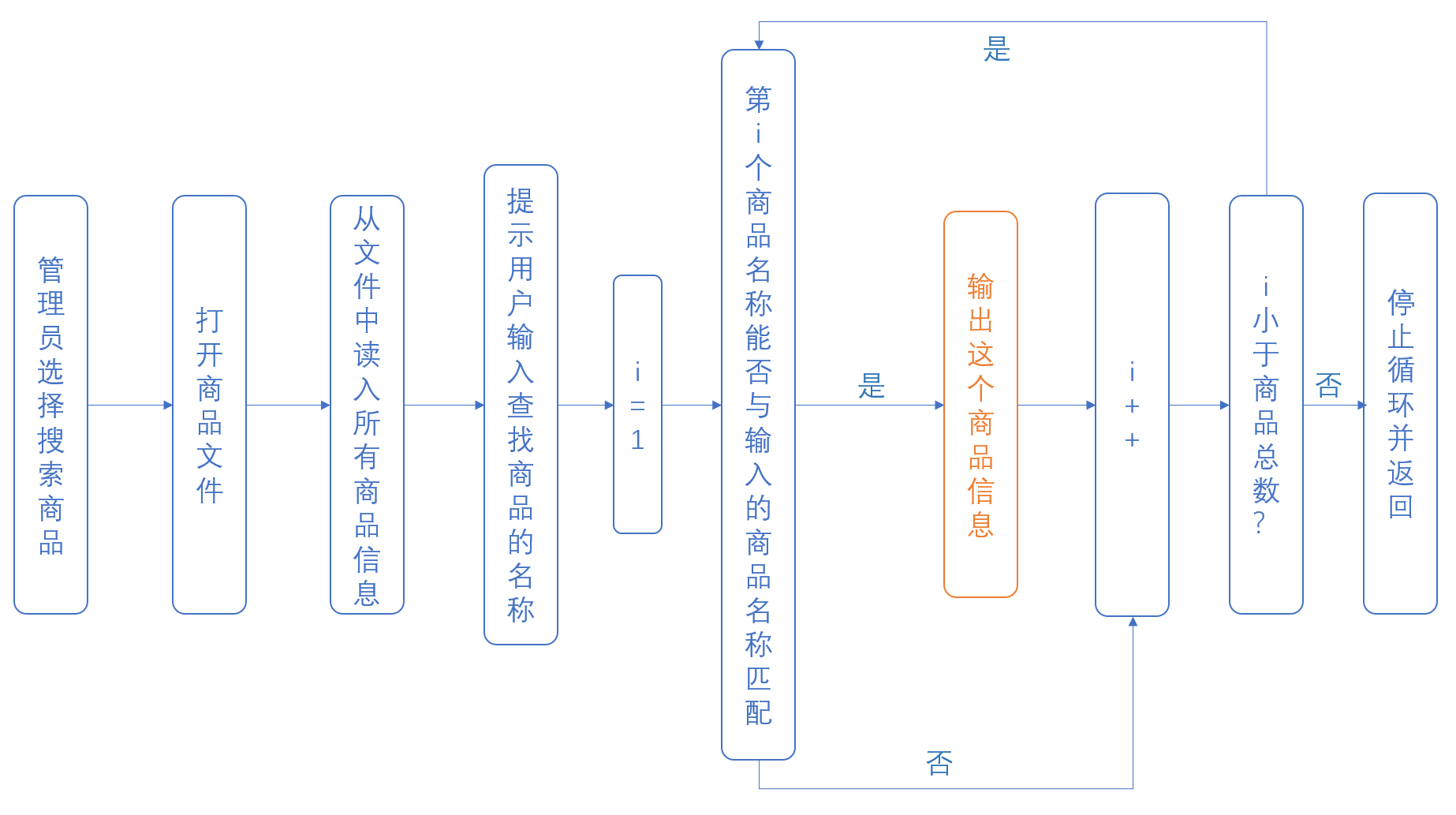
在用户选择管理员登录选项后，先提示用户输入用户名并对用户名进行判定。若用户名错误，则返回初始界面。若用户名输入正确，则提示用户输入密码，并仿照上面进行判定。若用户名密码均正确，则为登陆成功，可以开始执行管理员的一系列功能。

以下为查看所有商品的详细设计思路：



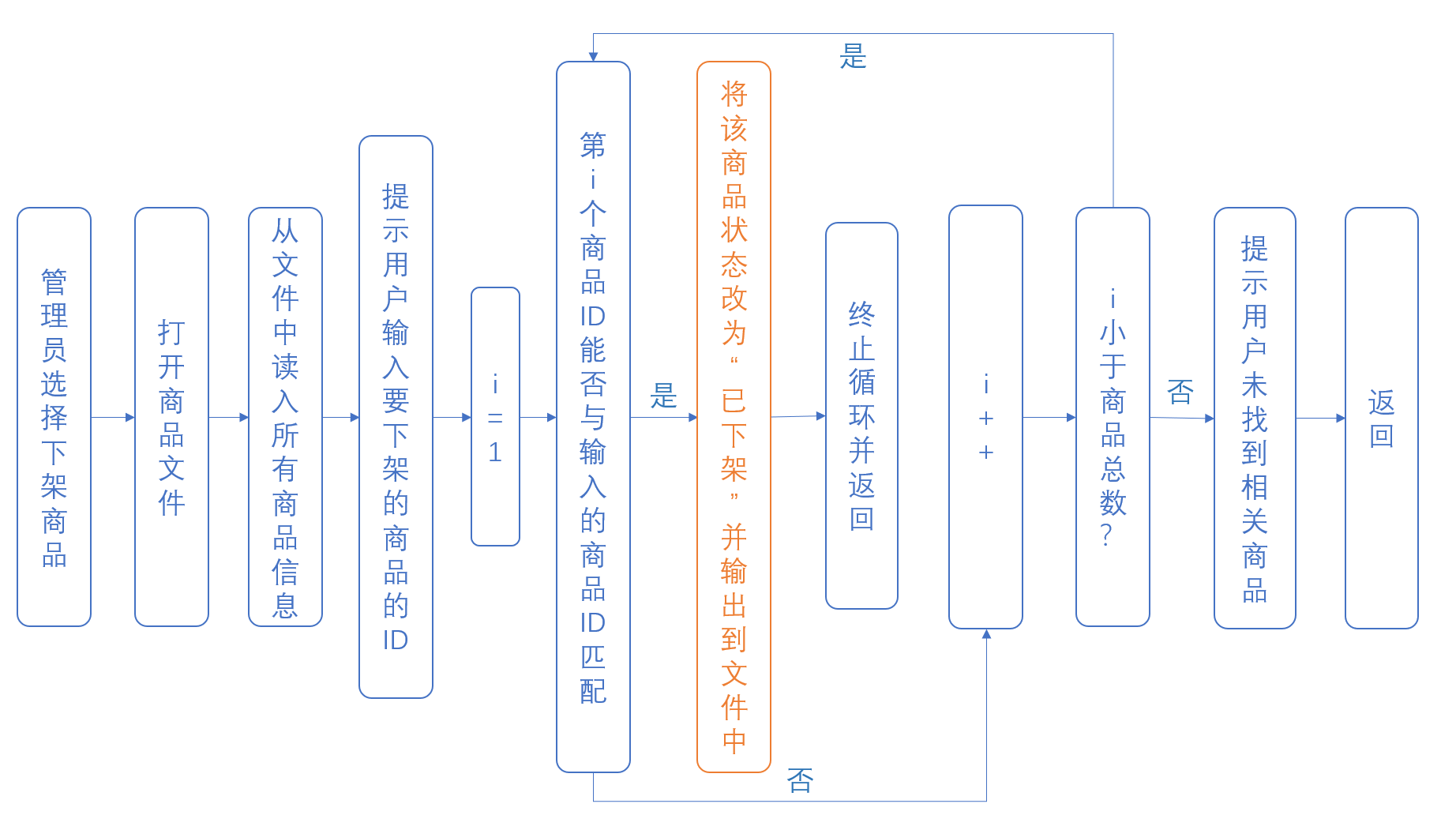
管理员选择查看所有商品后，先打开商品文件，将其中的数据全部读取出来存储到数组中，在将数组中的数据通过循环全部输出展示出来，最后执行返回操作。

以下为搜索商品操作的详细设计思路：

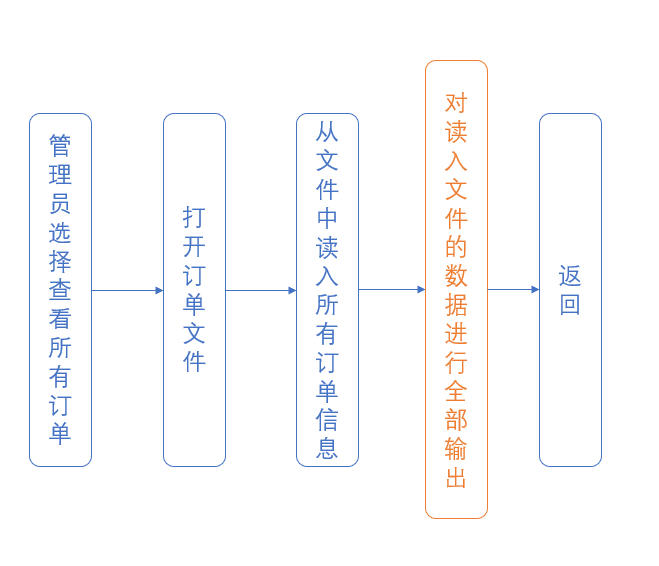


在用户选择搜索商品后，首先打开商品文件并读入其中的所有数据，提示用户输入待查找的商品名称。之后对所有读入的商品名称进行循环匹配，将所有可以匹配的商品信息全部输出。

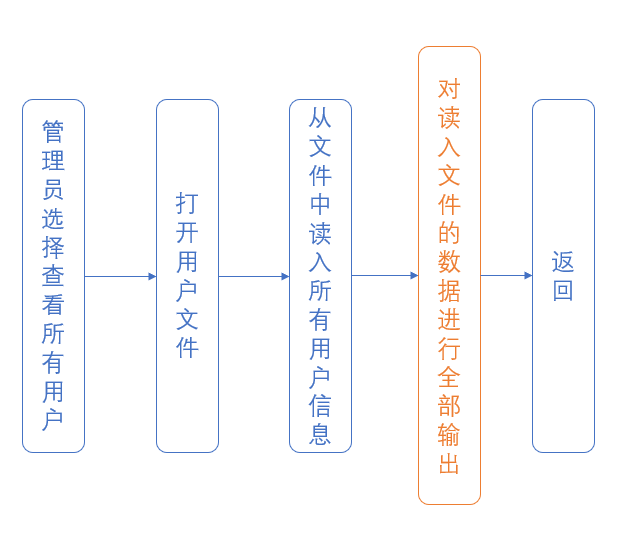
以下为下架商品的详细设计思路：



当管理员选择下架商品时，先打开商品文件并读入所有商品的信息，然后提示管理员输入要下架的商品ID。在程序中对要下架的商品ID进行循环匹配，直到找到要下架的商品。这时将该商品信息中的状态改为“已下架”，并对文件中的信息进行更新。若遍历所有信息仍未找到要下架的商品信息，则提示用户并返回上一层级。

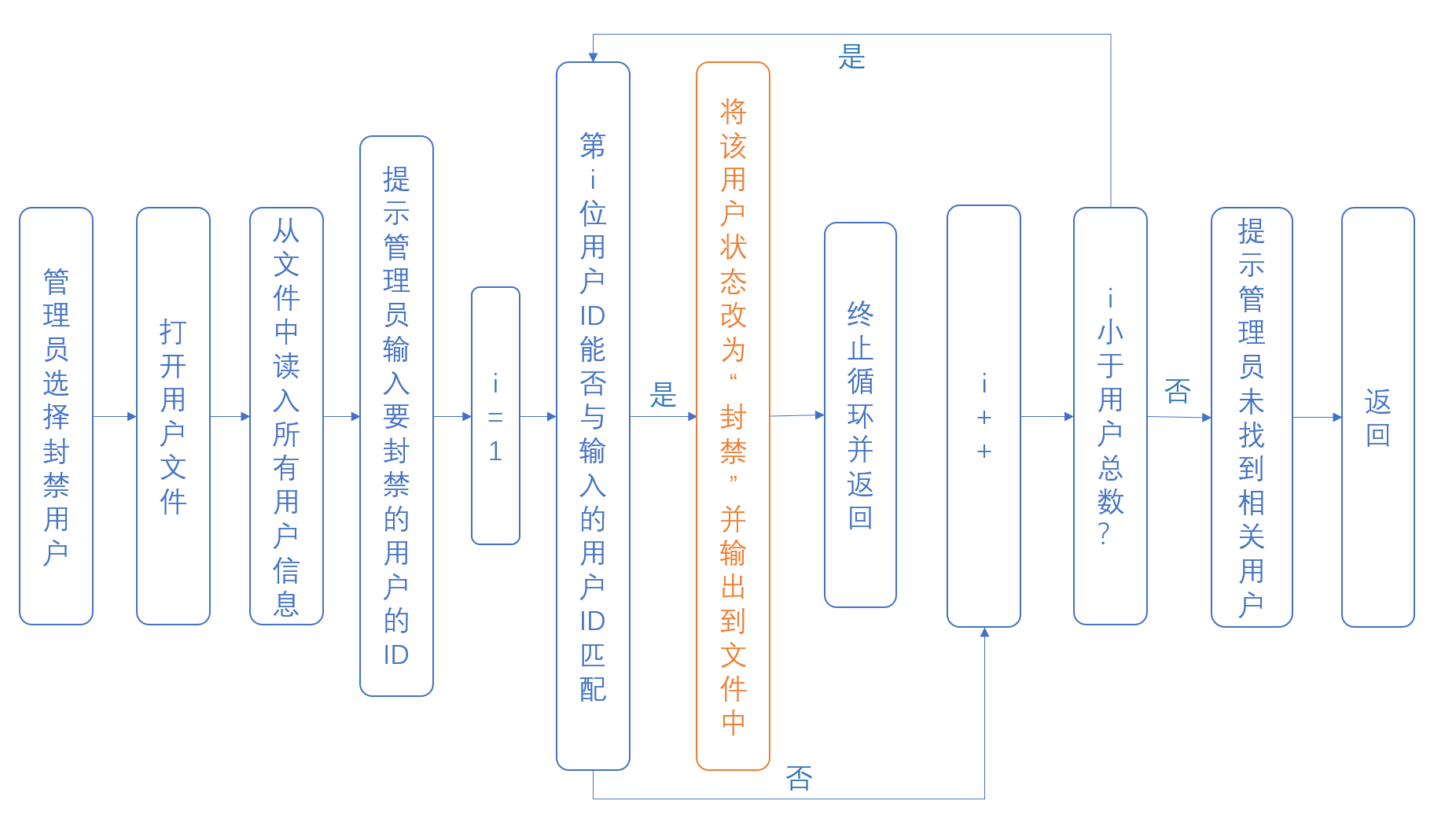
以下为查看所有订单操作的详细设计思路：  


管理员选择查看所有订单后，先打开订单文件，将其中的数据全部读取出来存储到数组中，在将数组中的数据通过循环全部输出展示出来，最后执行返回操作。

以下为查看所有用户的详细设计思路：  


管理员选择查看所有用户后，先打开用户文件，将其中的数据全部读取出来存储到数组中，在将数组中的数据通过循环全部输出展示出来，最后执行返回操作。

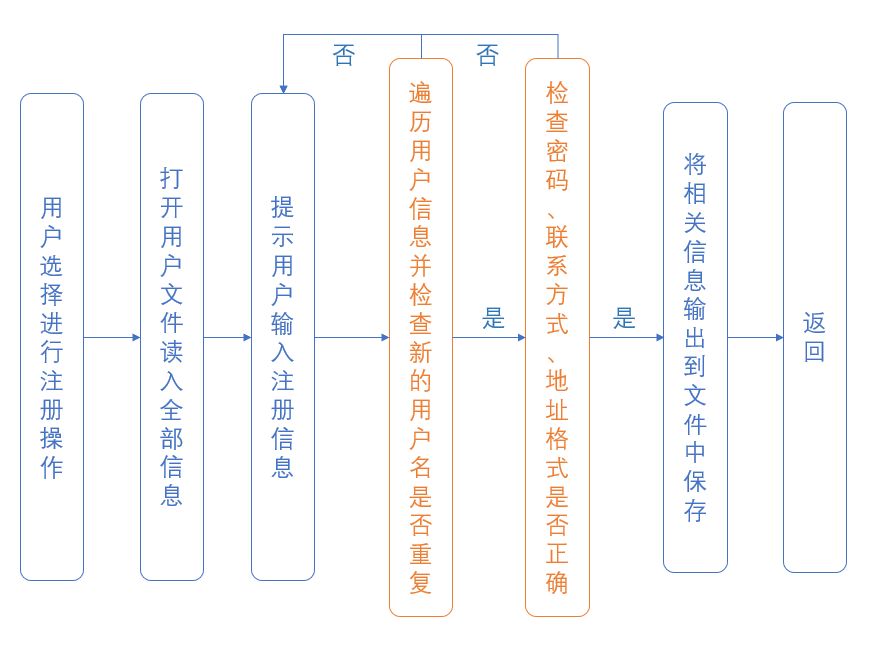
以下为封禁用户操作的详细设计思路：

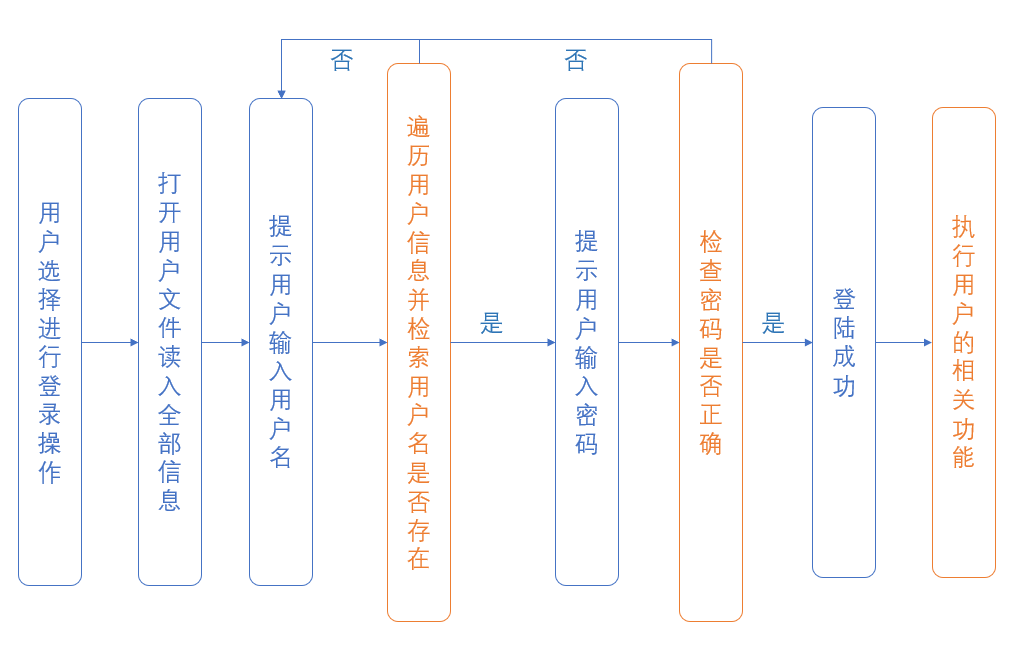
  
当管理员选择封禁用户时，先打开用户文件并读入所有用户的信息，然后提示管理员输入要封禁的用户ID。在程序中对要风景的用户ID进行循环匹配，直到找到要封禁的用户。这时将该用户信息中的状态改为“封禁”，并对文件中的信息进行更新。若遍历所有信息仍未找到要风景的用户信息，则提示管理员并返回上一层级。

**日期：12月5日**

**1.实验内容：详细设计（用户注册与登录部分）**

以下是用户注册的详细设计思路：

  
若用户选择进行注册操作，则先打开用户文件并读入所有数据，在提示用户对注册信息进行输入。若输入的信息不合法，则注册失败。若信息均合法，则注册成功，并将相关信息输出到文件中储存。最后执行返回操作。

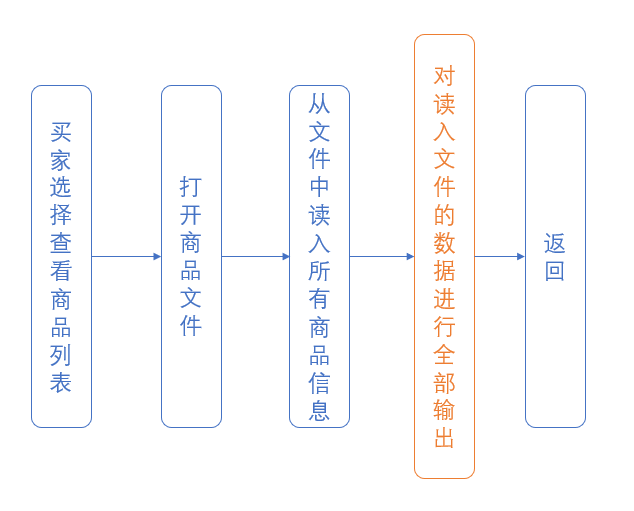
以下是用户登录的详细设计思路：  
  
若用户选择进行登录操作，则先打开用户文件并读入所有数据，在提示用户对输入用户名。

遍历所有的用户，找到匹配的用户名后提示用户输入密码。若输入的密码正确，则登陆成功。否则登录失败。若遍历所有用户信息也未能找到相匹配的用户名，则提示用户的用户名输入错误，并执行返回操作。若登陆成功，则可执行后续的用户相关操作。

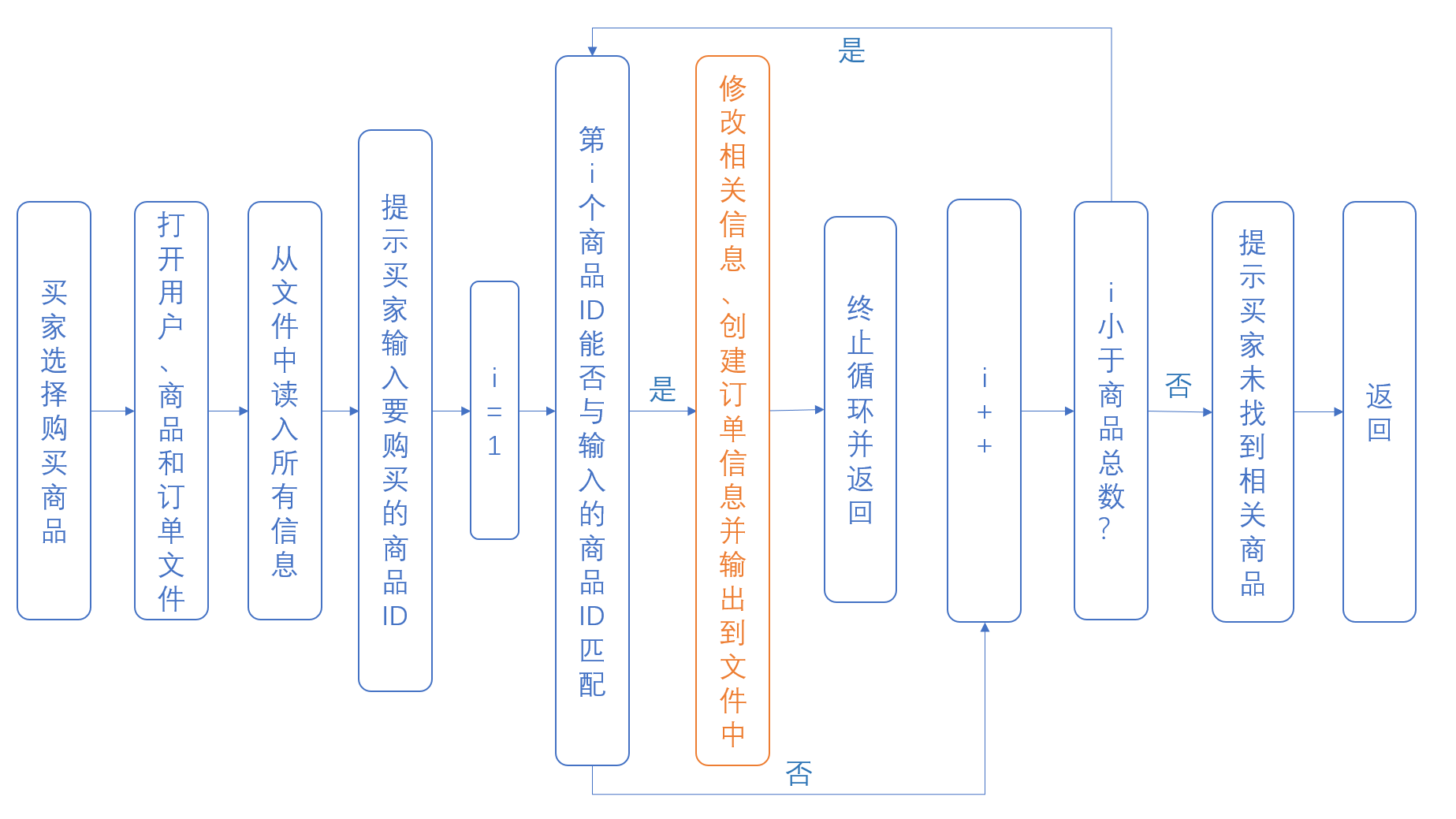
**日期：12月6日**

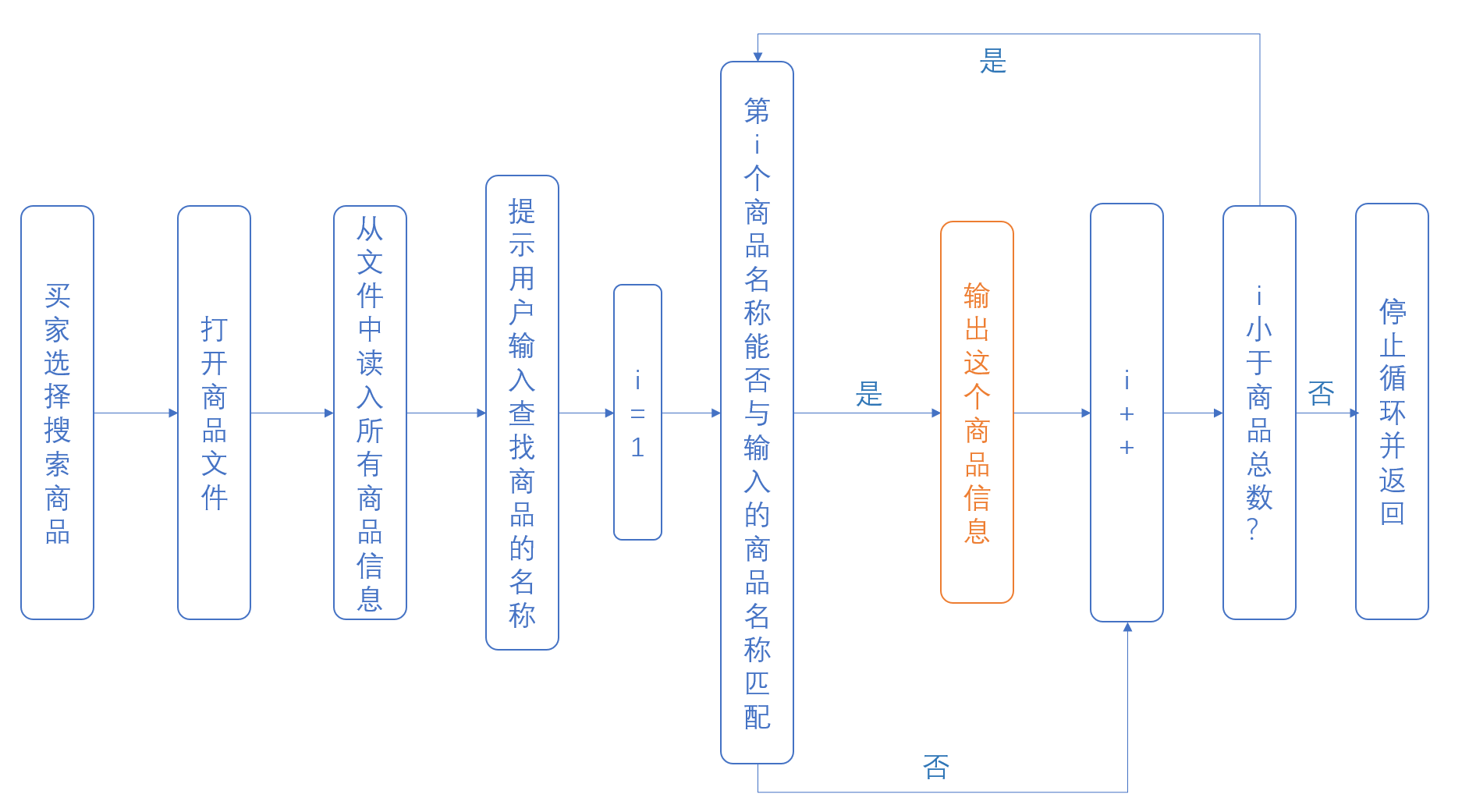
**1.实验内容：详细设计（买家部分）**

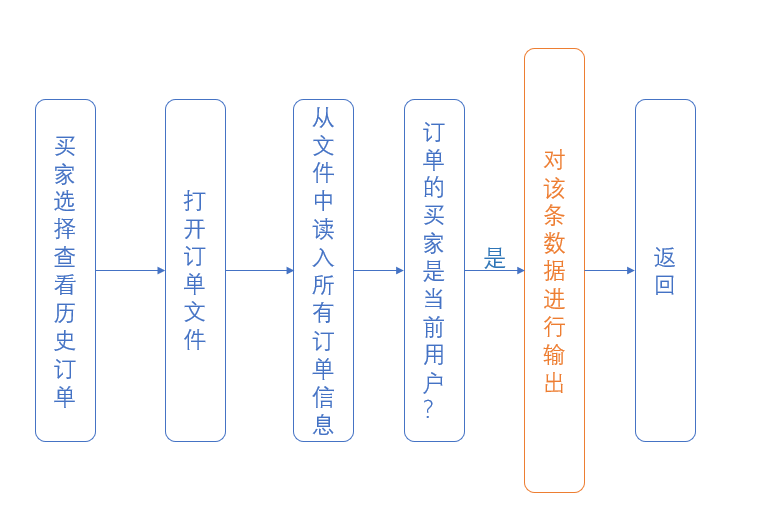
以下是买家查看商品列表的详细设计思路：

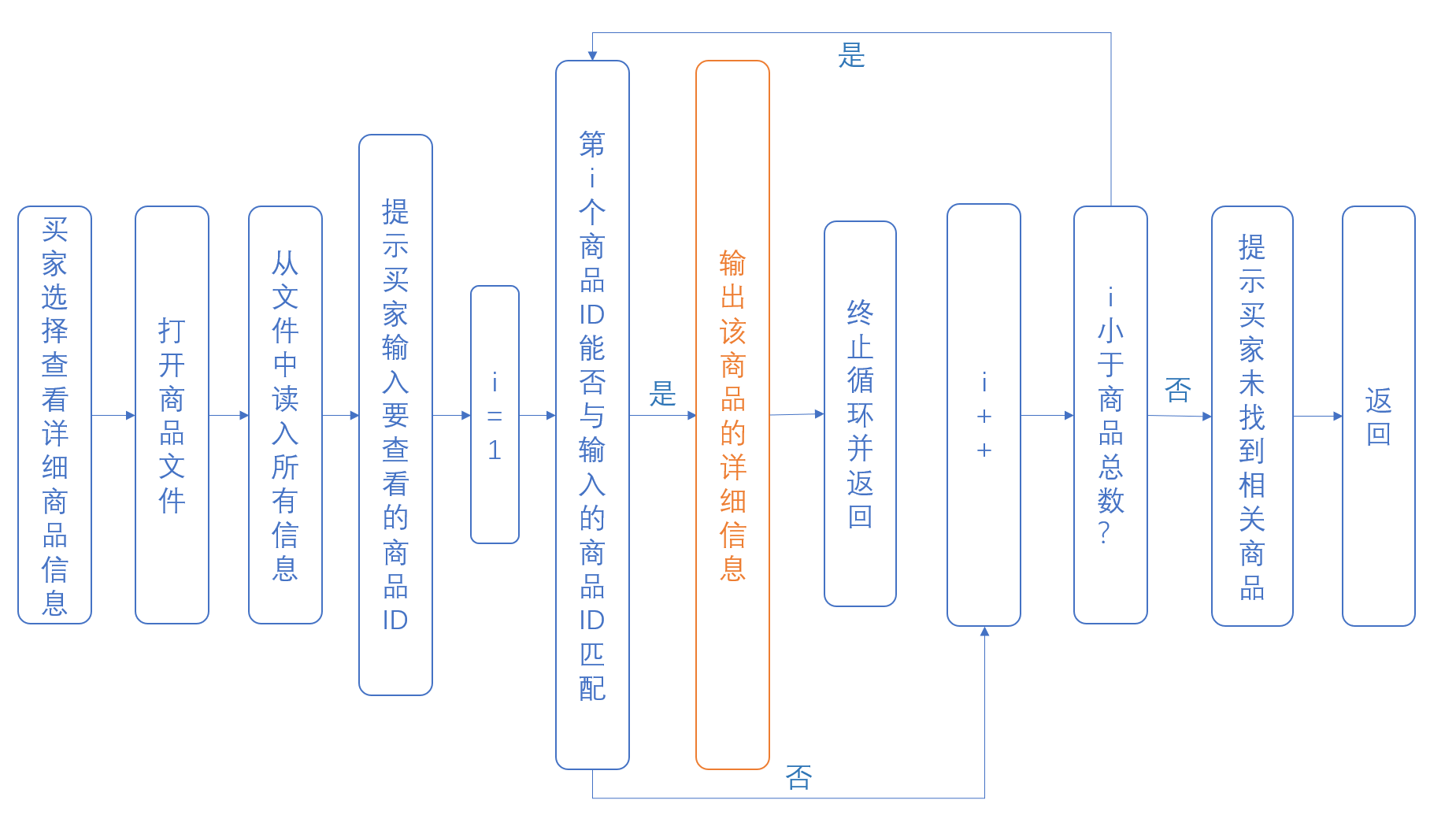


买家选择查看所有商品后，先打开商品文件，将其中的数据全部读取出来存储到数组中，在将数组中的数据通过循环全部输出展示出来，最后执行返回操作。

以下是购买商品的详细设计思路：  


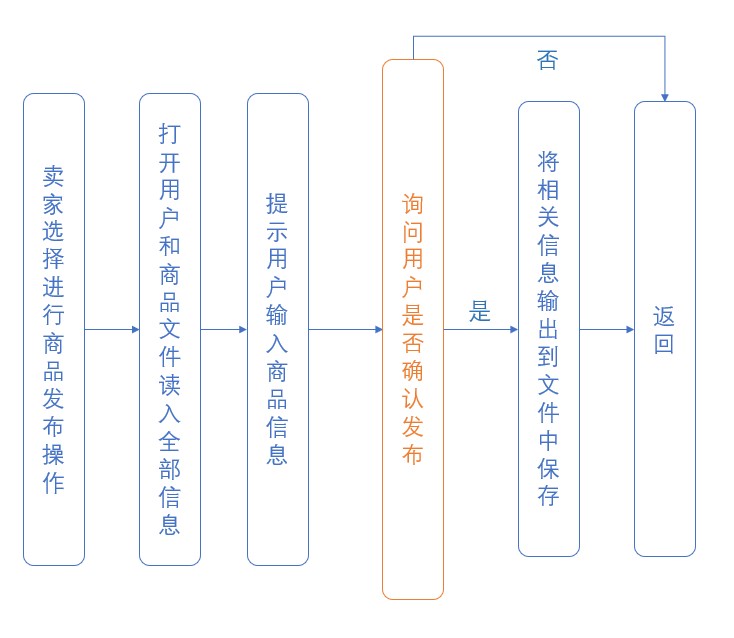
以下是搜索商品的详细设计思路：  


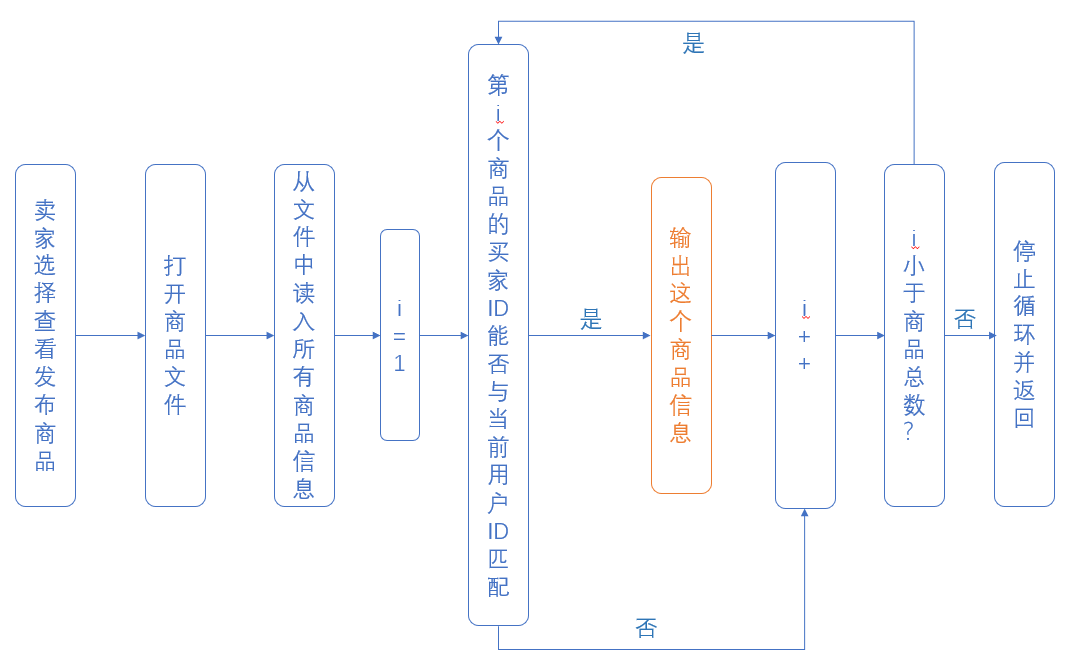
以下是买家查看历史订单的详细设计思路：  


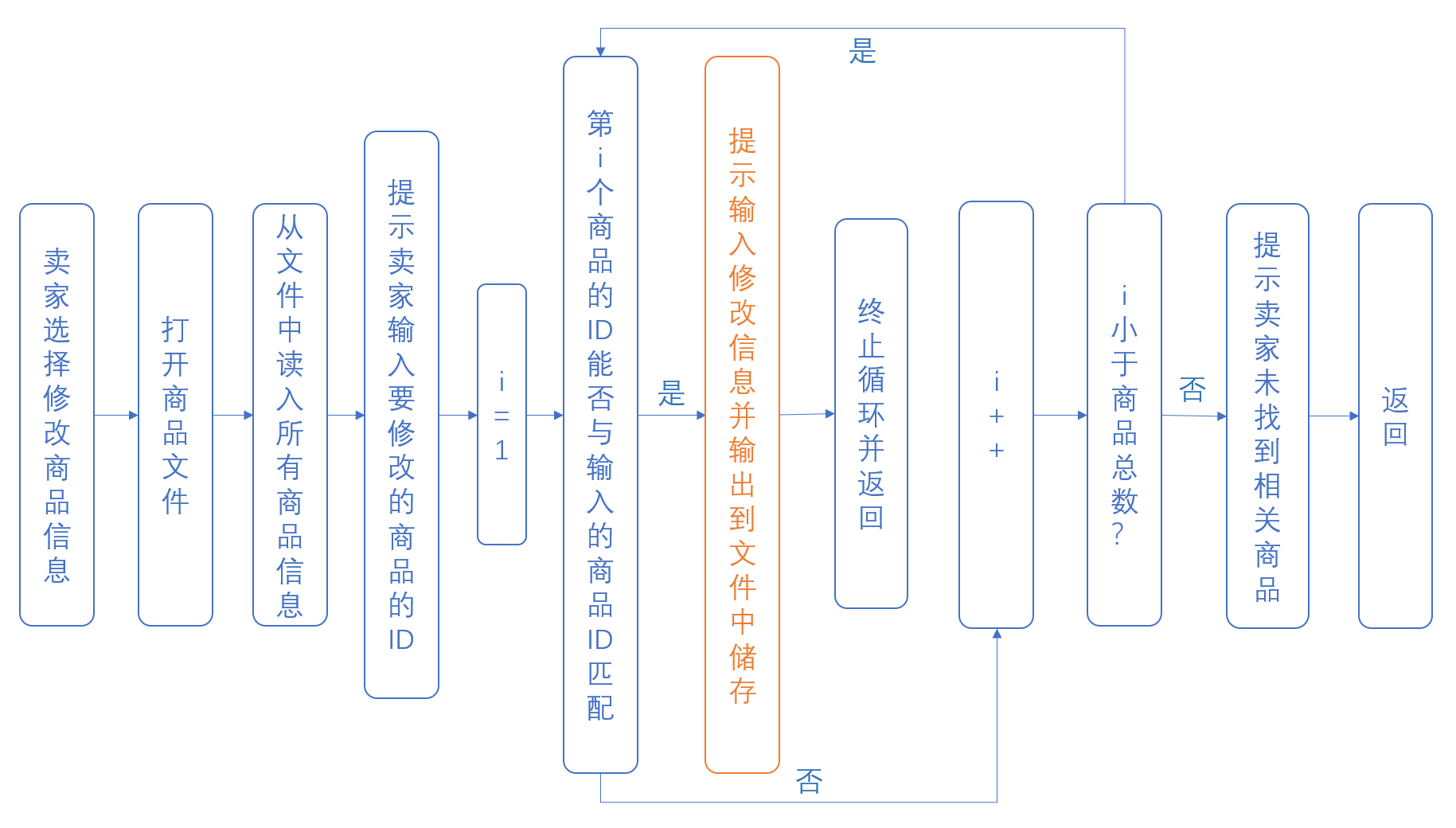
以下是买家查看详细商品信息的详细设计思路：  


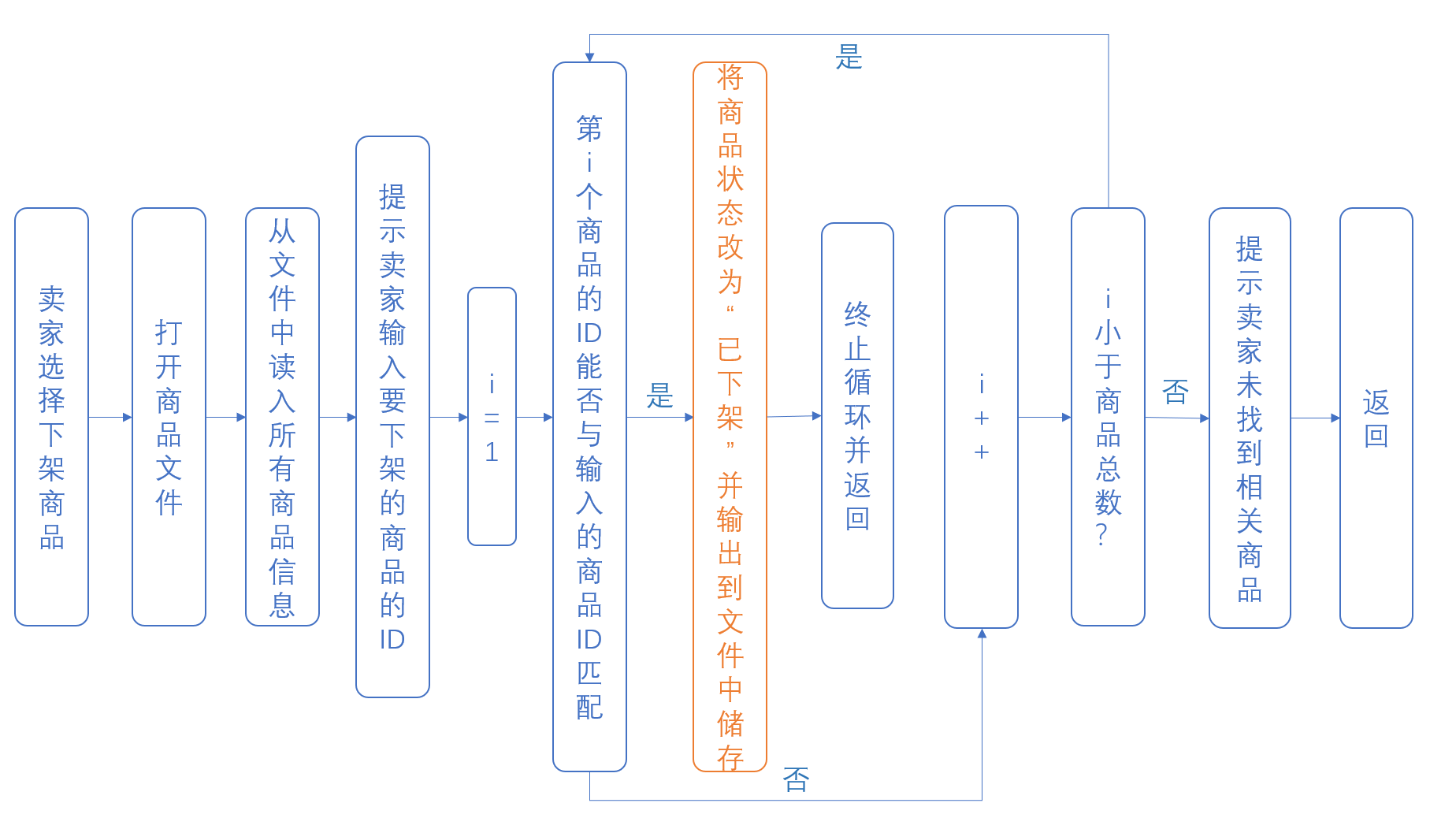
**日期：12月7日**

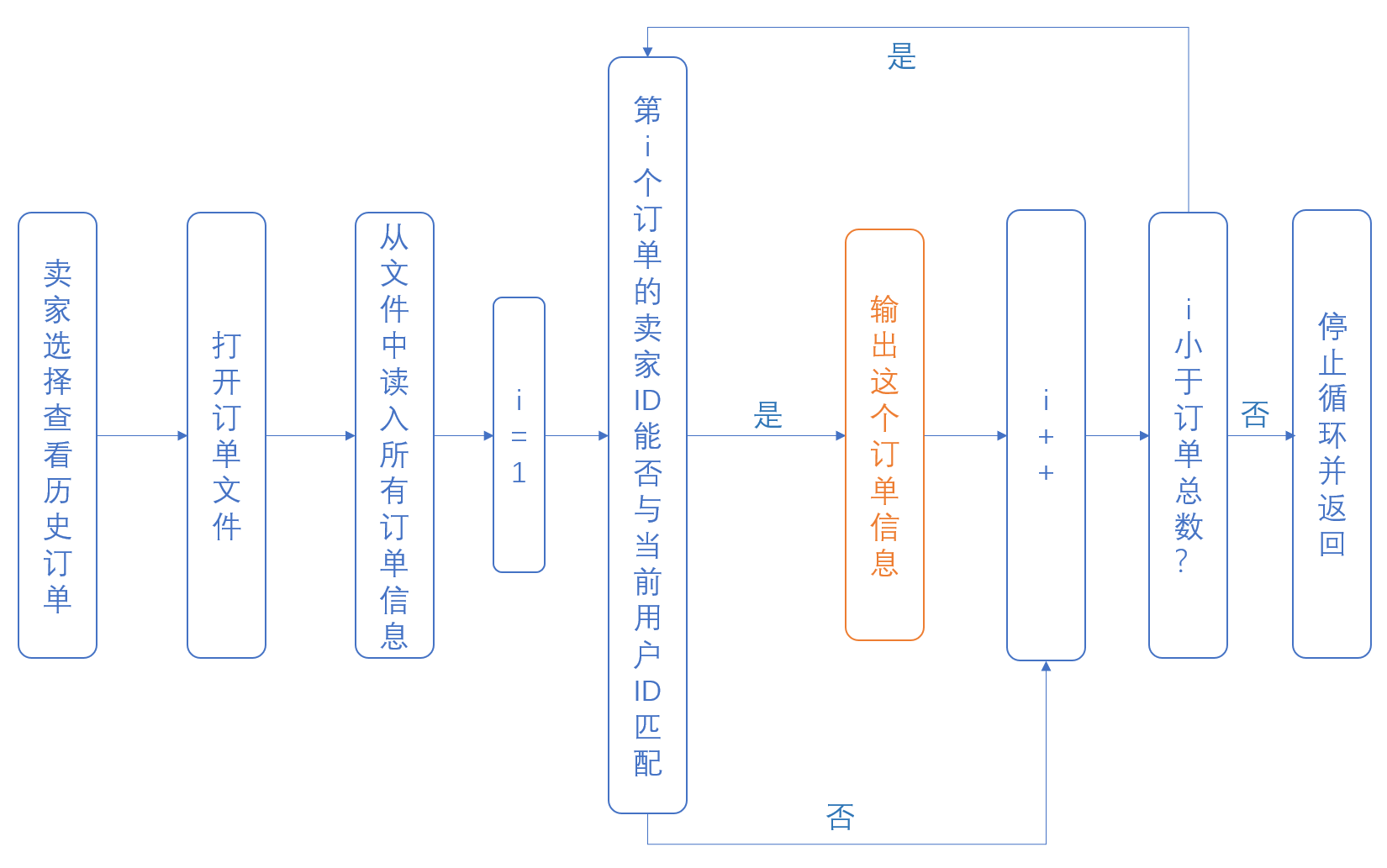
**1.实验内容：详细设计（卖家部分）**

以下是卖家发布商品部分的详细设计思路：  


以下是卖家查看发布商品功能的详细设计思路：  


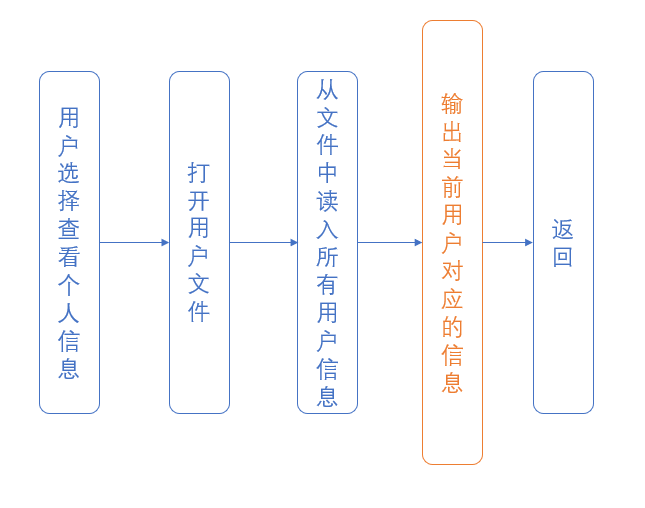
以下是卖家修改商品信息的详细设计思路：  


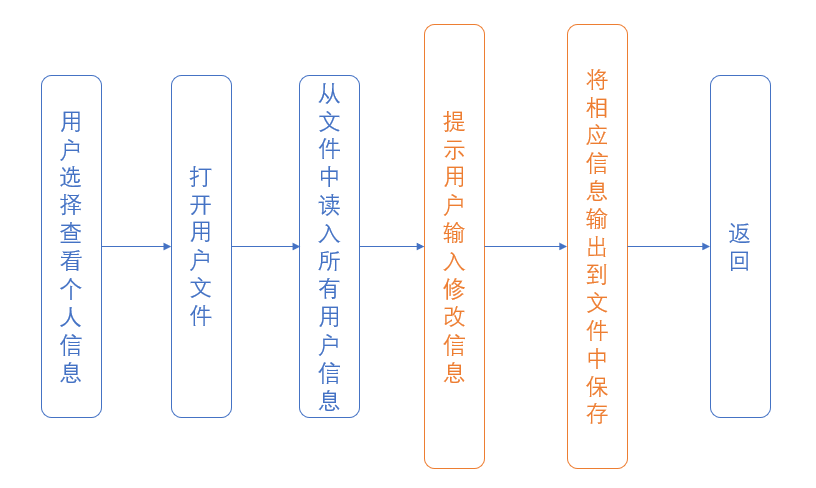
以下是卖家下架商品信息的详细设计思路：  


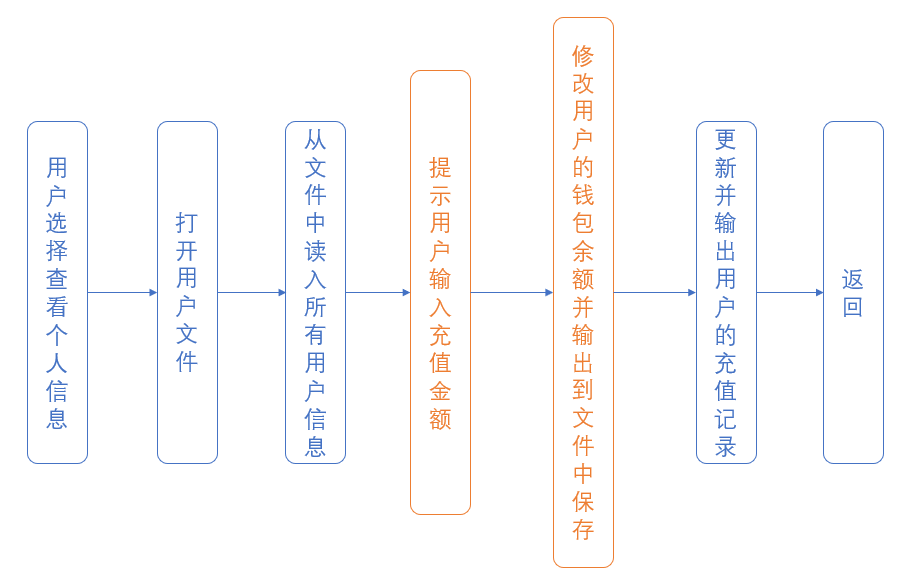
以下是卖家查看历史订单的详细设计思路：  


**日期：12月8日**

**1.实验内容：详细设计（用户信息管理部分）**

以下是用户查看个人信息的详细设计思路：  


以下是用户修改信息的详细设计思路：  


以下是用户充值功能的详细设计思路：  


**日期：12月9日**

**1.实验内容：编程实现**

管理员的查看商品列表、查看所有用户、查看所有订单，买家的查看商品列表、查看历史订单，以及卖家的查看发布商品、查看历史订单功能大体相近，不妨放在一起进行编码实现。

首先应当调用<fstream>头文件中的“ifstream”来打开对应的文件，用定义的in\_file来从文件中抽取数据存放在我已经定义好的结构数组中。在这之后根据相应的筛选条件输出相应的数据即可。最后用“return”来进行返回上一层级的操作。

应当注意的是，上述管理员功能并没有筛选条件，而买家查看商品列表的条件则是(strcmp(item[i].status,”销售中”)==0)，而买家查看历史订单、卖家查看已发布商品、卖家查看历史订单的筛选条件则都是相应订单或商品的卖家或买家ID与当前用户的ID相同。

这几个功能调用的函数分别为manager01，manager04，manager05，buyer01，buyer04，seller02，seller05。函数均为void类型，没有返回值和输入值，通过switch语句完成调用。

管理员的搜索商品和买家的搜索商品大体相近，都是根据用户输入的名称进行循环匹配，放在一起实现。

首先依然是打开文件抽取数据，与上文相同，不再赘述。其次则是提示用户输入名称以后根据名称进行循环匹配。利用for(int i=0;i<itemNumber;i++)完成循环，如果满足if(strcmp(item[i].name,name)==0)则对该组数据进行输出。需要注意的是用户查找商品时还应当增加一个商品状态是销售中的判断：if(strcmp(item[i].status,”销售中”)==0)。

该部分对应的函数为manager02，buyer03。函数均为void类型，没有形参和返回值，通过switch语句调用。

管理员的下架商品，封禁用户、买家的购买商品、查看商品的详细信息、卖家的下架商品、修改商品信息功能相近，都是根据用户输入的ID进行查找。

首先仍然是打开文件读入数据。然后提示用户输入数据。利用for(int i=0;i<n;i++)来实现循环。如果满足if(strcmp(item[i].ID,ID)==0)则直接利用break语句跳出循环，然后对第i个对象进行一系列修改或者是直接输出。如果需要向文件中输出修改后的内容，则利用“ofstream”来定义一个out\_file功能，向文件中依次输入信息。

该部分函数分别为manager03、manager04、buyer02、buyer05、seller03、seller04。以上函数均为void类型，没有形参和返回值，通过switch语句调用。

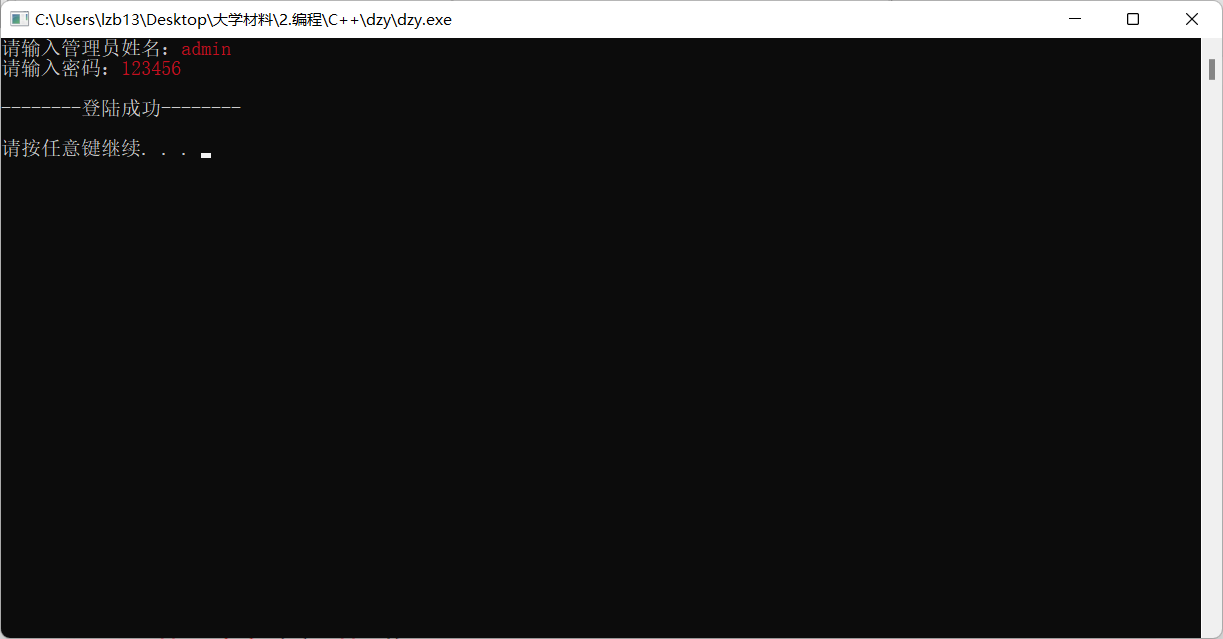
卖家发布商品功能和用户注册功能大体相近，都是向文件中直接增添一组数据。

首先打开文件，将文件中已有的数据全部读入程序，以便后续输出。之后提示用户输入新的数据，新建一个结构体保存这个数据，之后连同之前读入的数据全部输出到文件中。

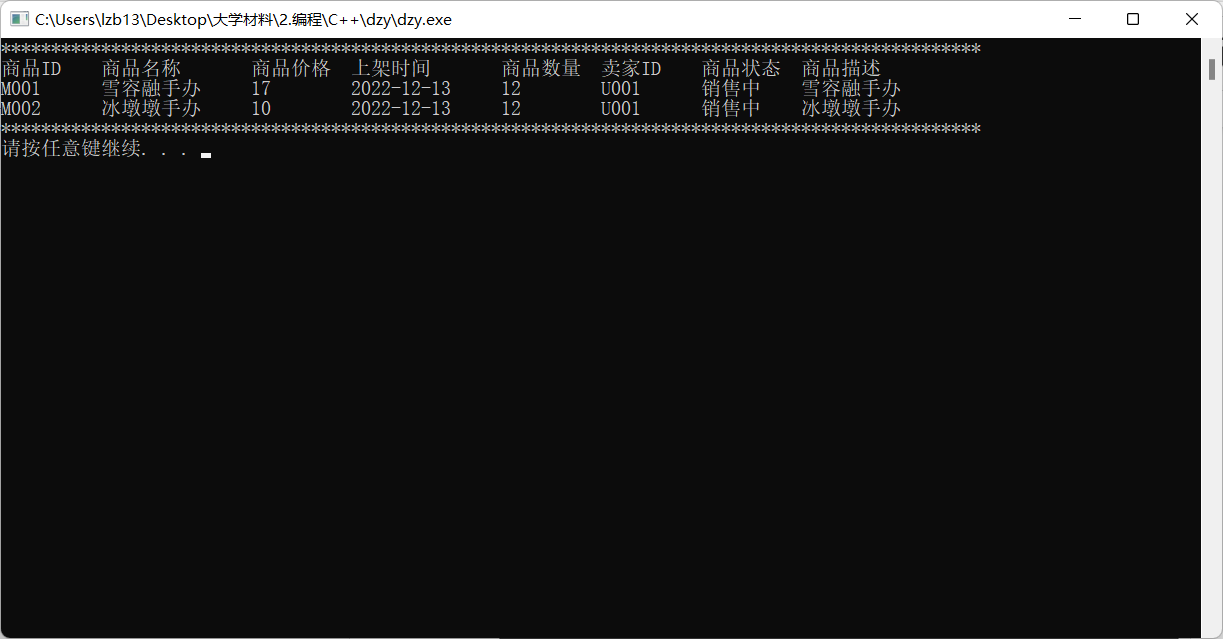
该部分调用的函数有seller01，userRegister。两个函数均为void类型，没有返回值，没有形参，通过switch语句调用。

**日期：12月10日**

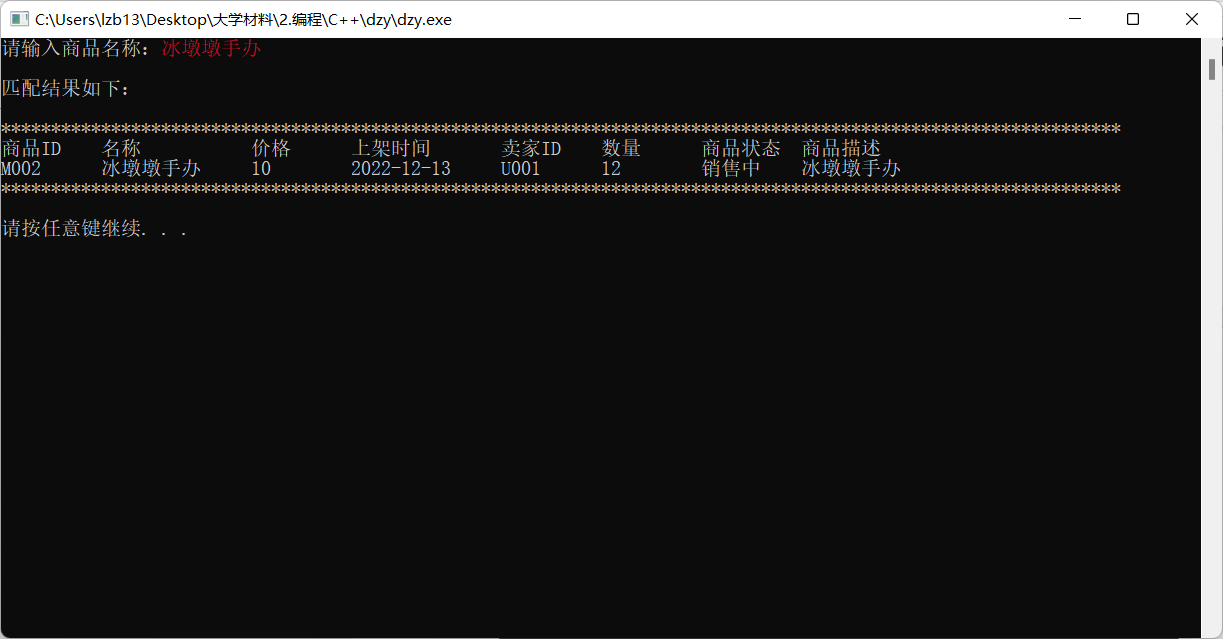
**1.实验内容：调试测试**



管理员的登录功能。



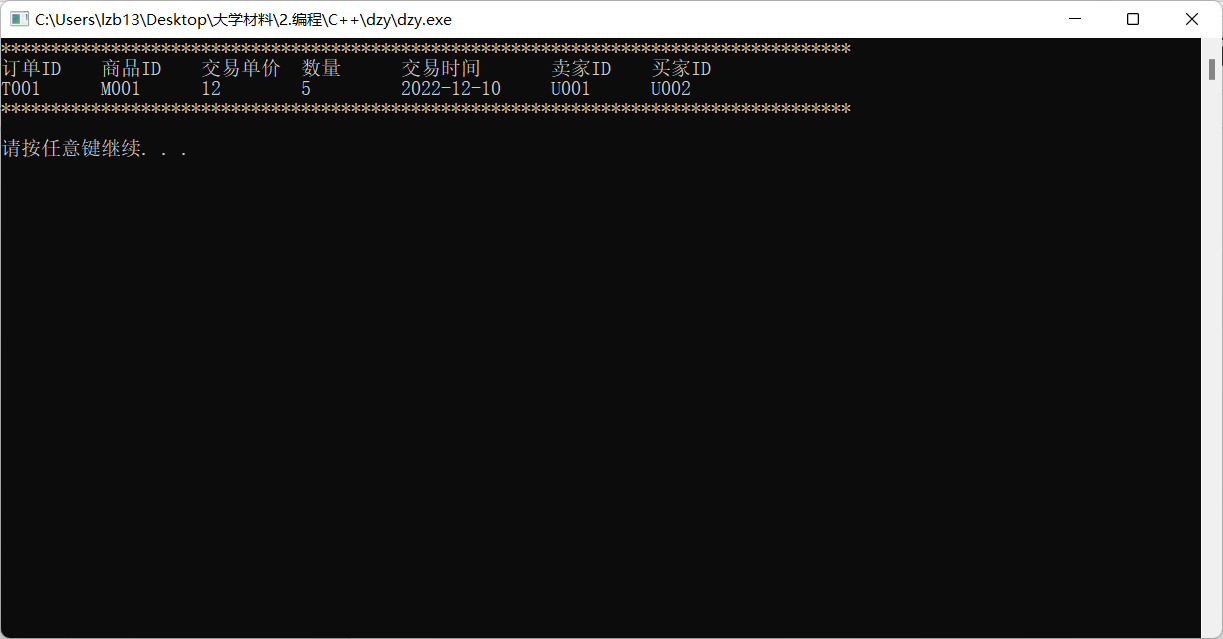
管理员的查看所有商品功能。



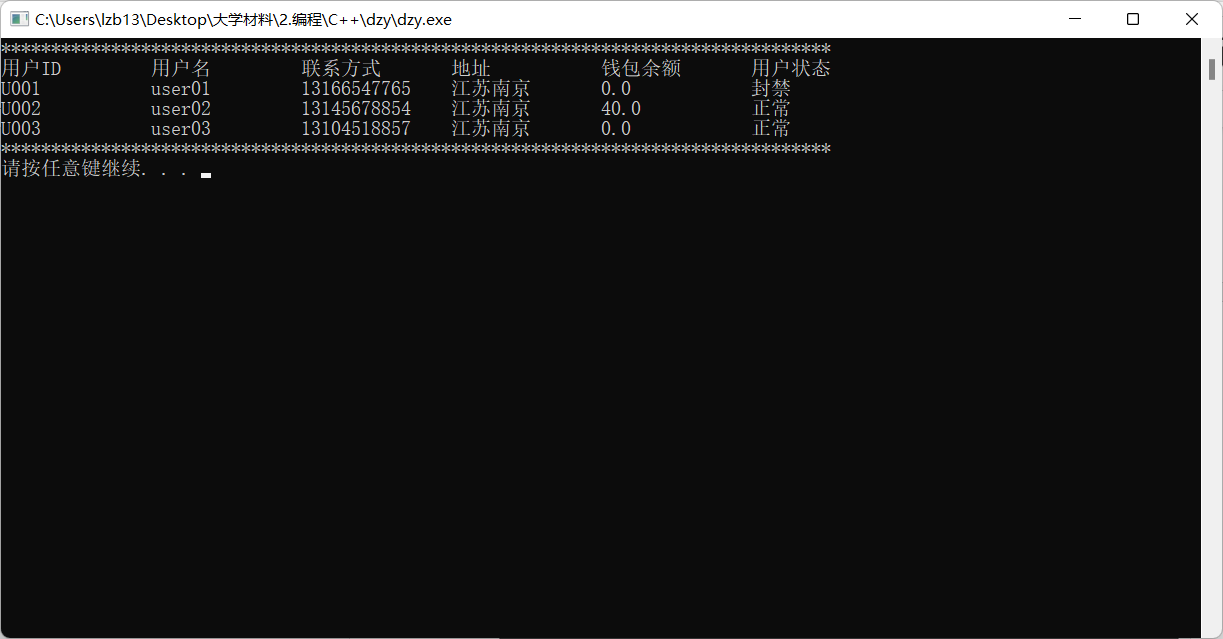
管理员的搜索商品功能。



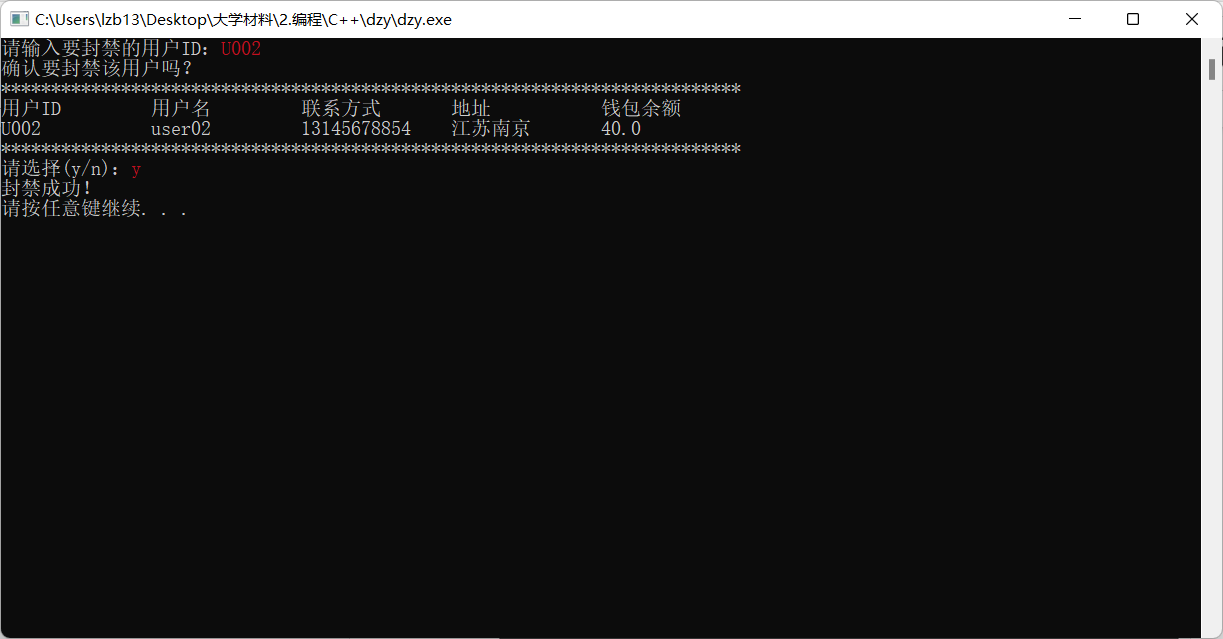
管理员的下架商品功能。



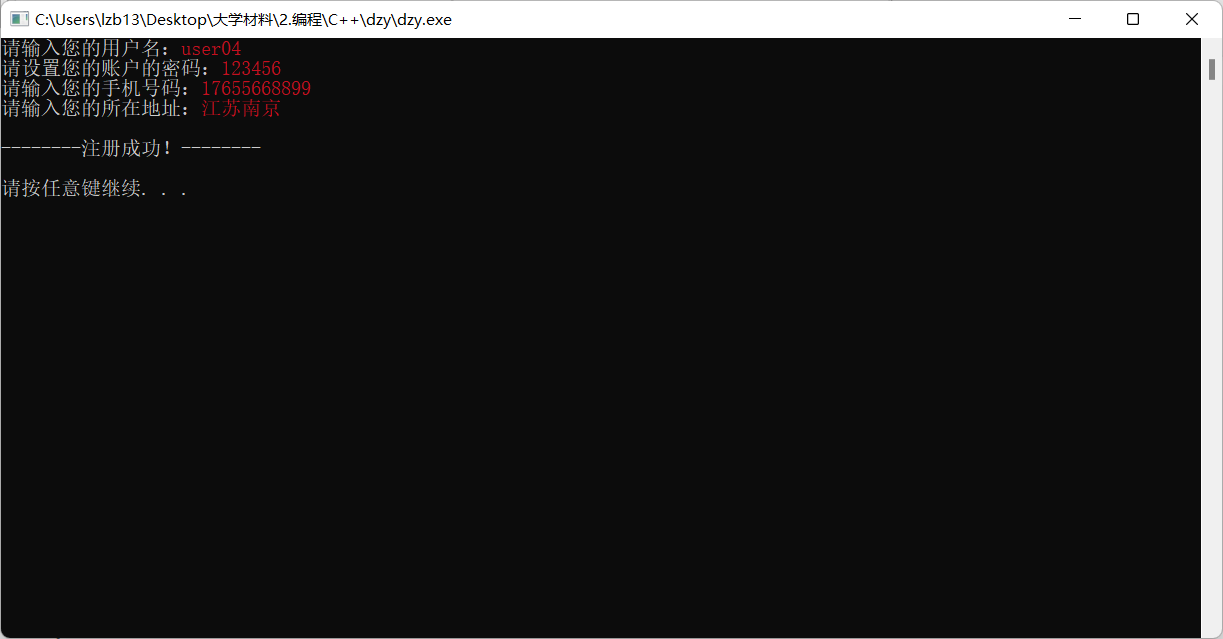
管理员的查看所有订单功能。



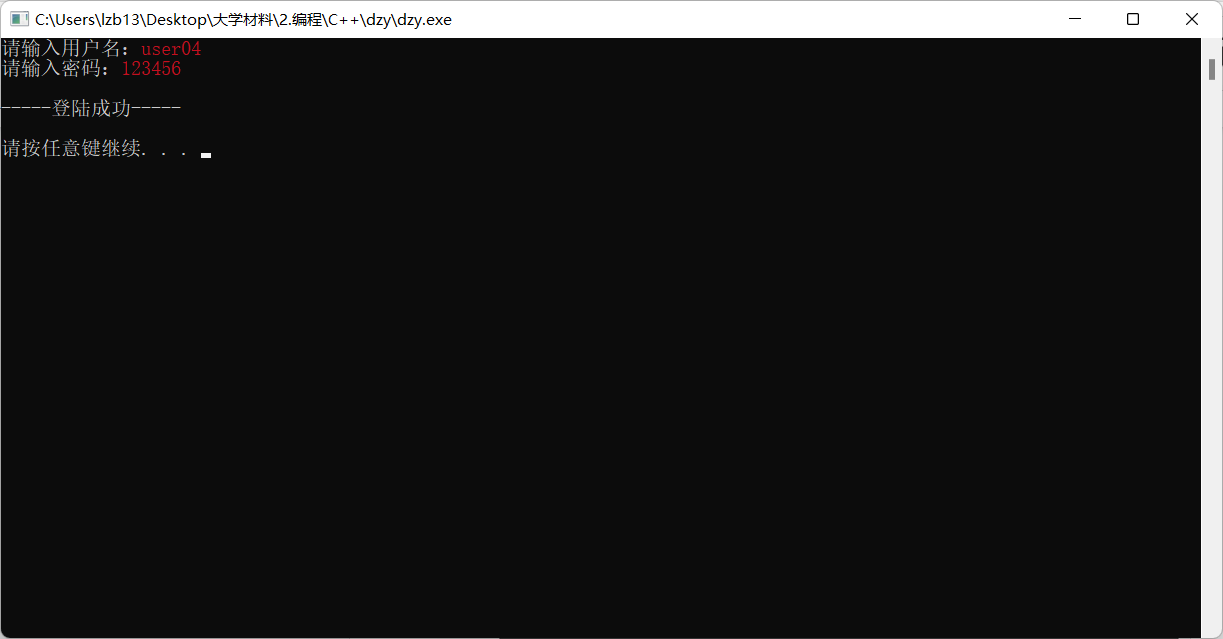
管理员的查看所有用户的功能。



管理员的封禁用户功能。



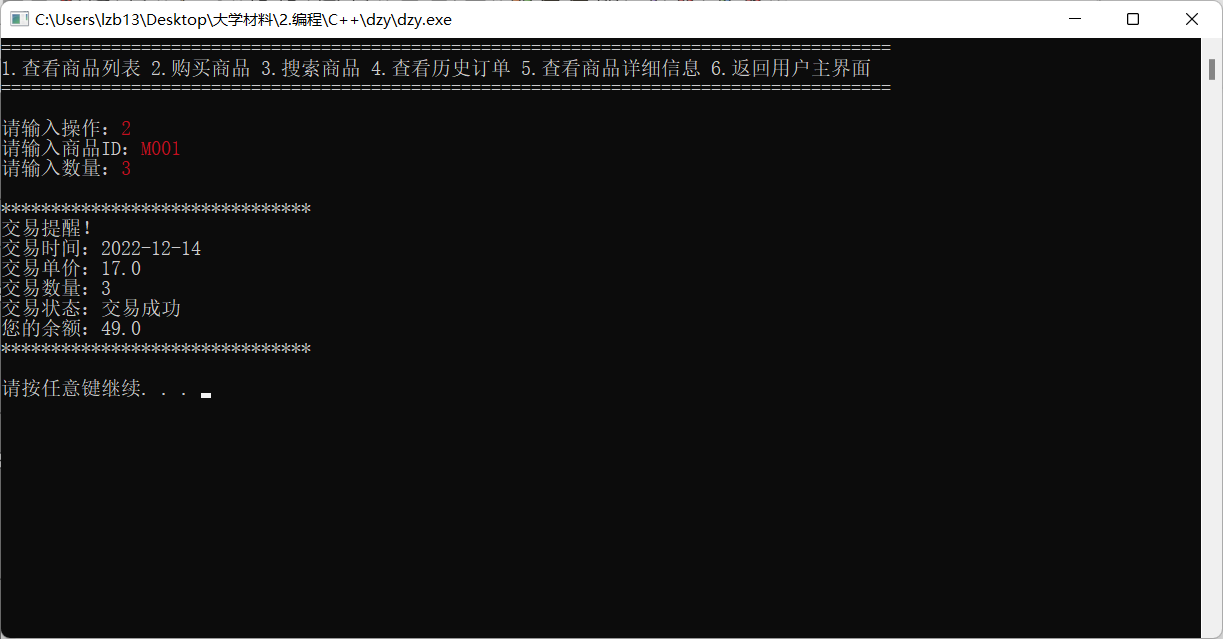
用户的注册功能。



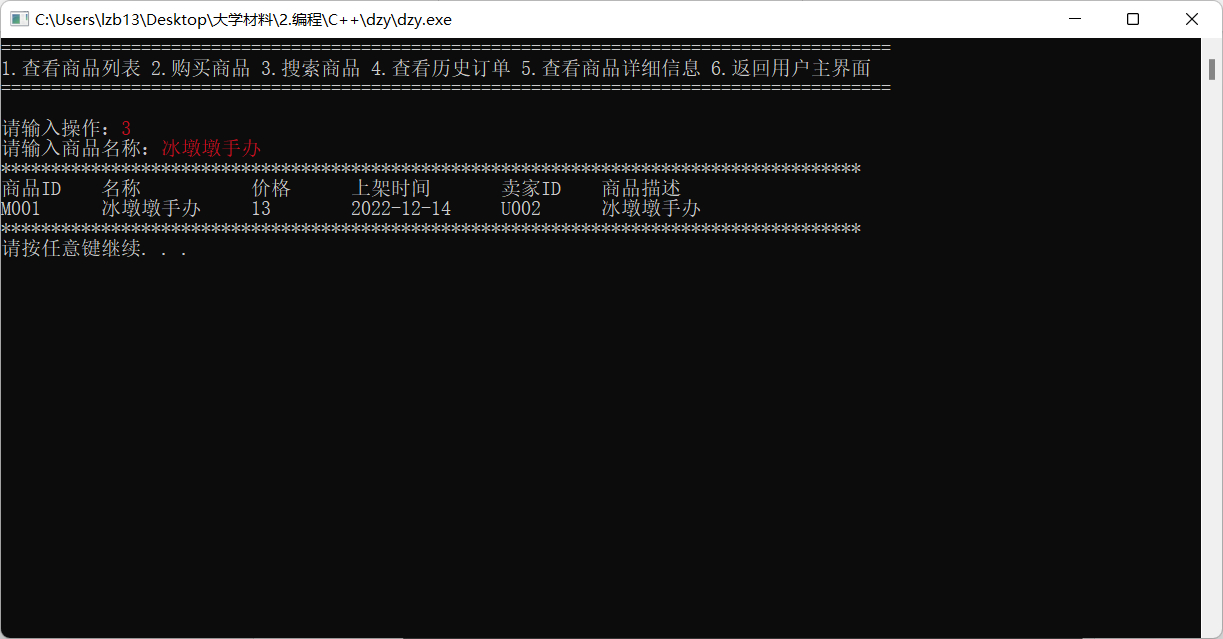
用户的登录功能。



买家的查看商品列表功能。



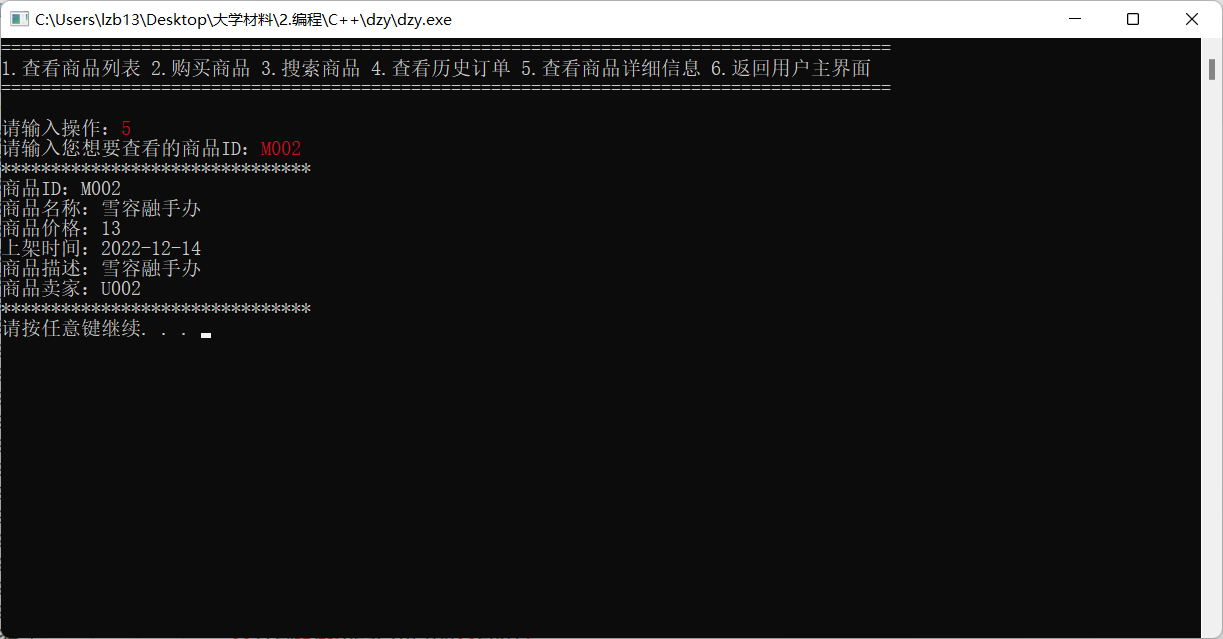
买家的购买商品功能。



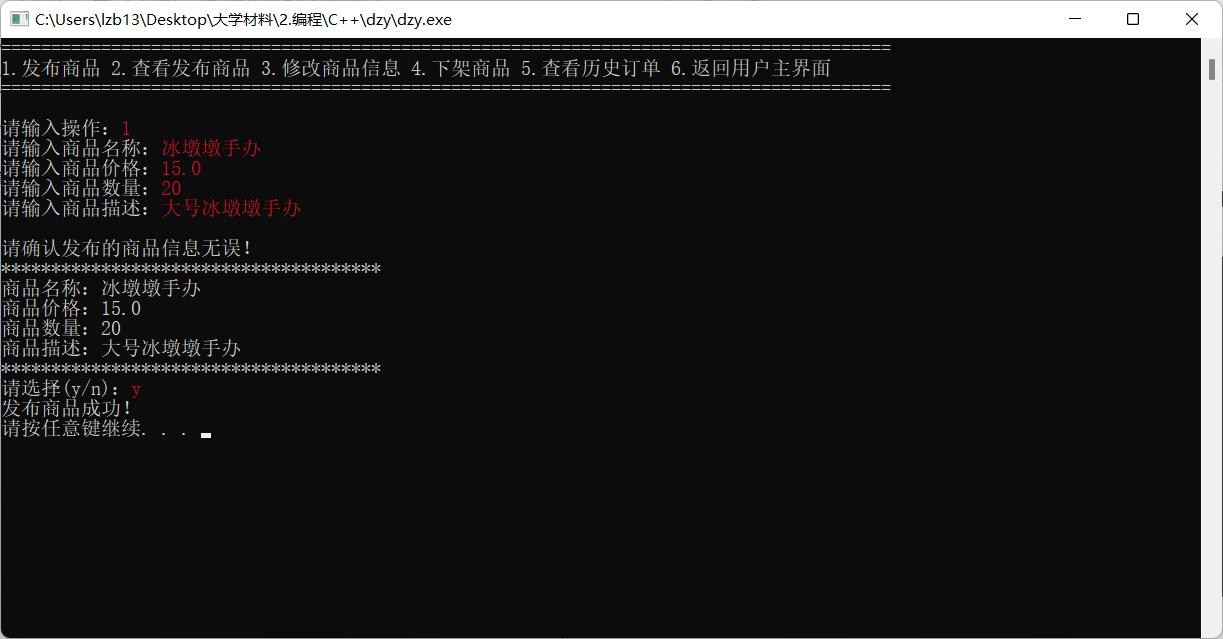
买家的搜索商品的功能。



买家的查看历史订单功能。



买家查看商品详细信息。



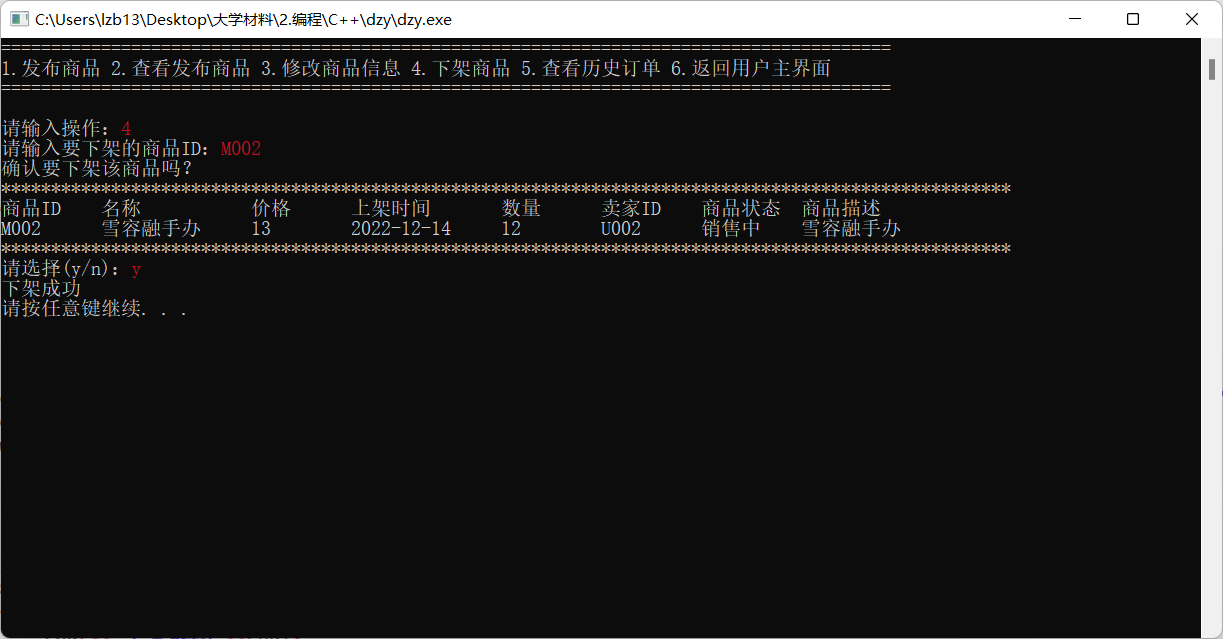
卖家发布商品功能。



卖家查看发布商品。



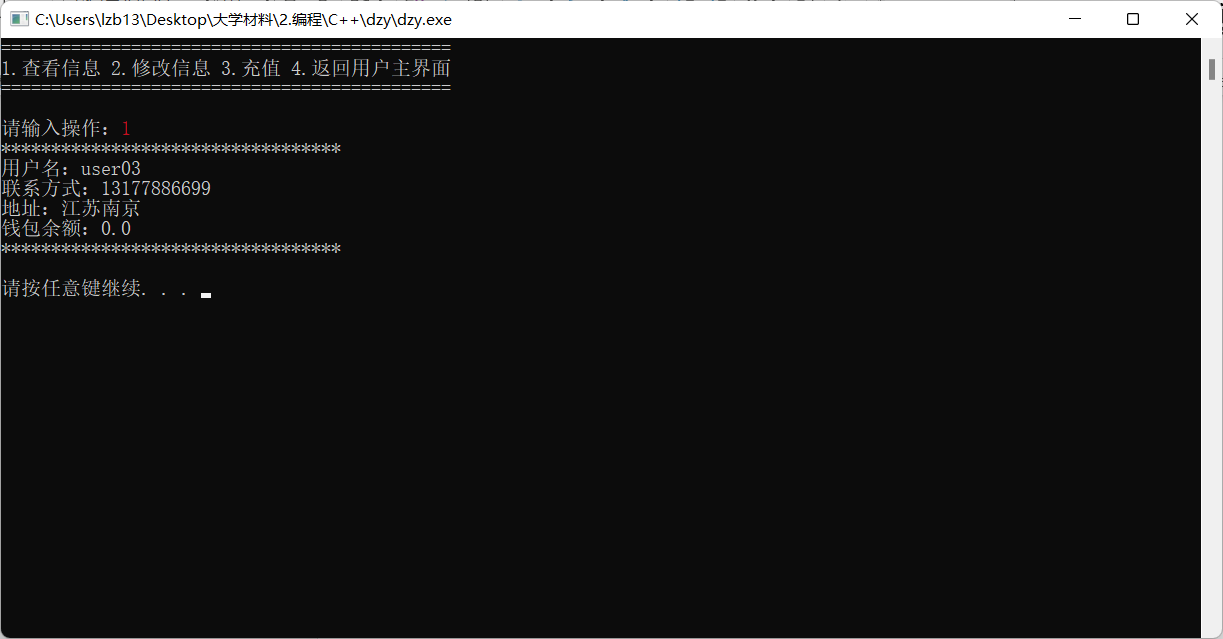
卖家修改商品信息。



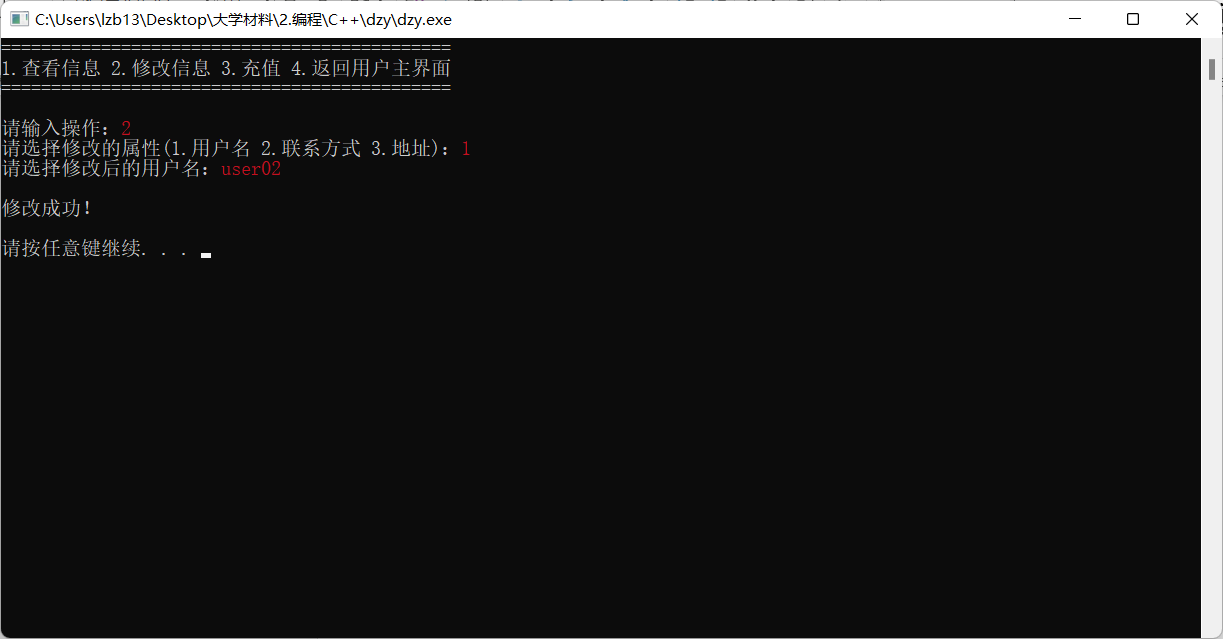
卖家下架商品。



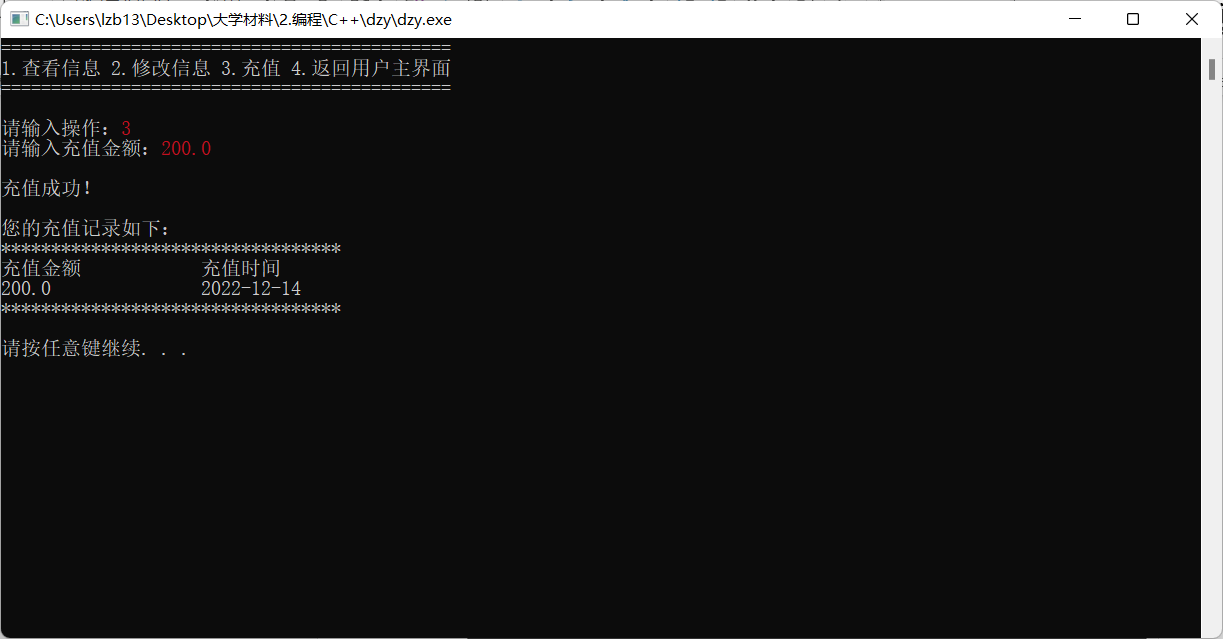
卖家查看历史订单。



用户查看个人信息。



用户修改个人信息。



用户充值。