# C语言程序设计:*六级背单词程序*

## 摘要

如今各种背单词APP层出不穷，这启发着我们自主运用C语言知识去设计一个背单词的程序。当然，设计这样一个程序仅仅运用课上所学习的知识是不够的，但是可以运用老师传授的内功心法去拓展学习开发本程序所需要的相关C语言知识。本程序和绝大多数背单词APP一样有学习功能和测试功能，主角为学生。此外还设计了管理者和老师这两个角色，每个角色都可以进行自己独特功能的操作。本程序在实现管理员管理程序、学生记单词、老师查看学生学习成果等基本功能的基础上，为了激发学生的学习兴趣，还开发了商店和贪吃蛇游戏功能。

## 简介与问题描述

我们的目标是开发一个可以存储用户信息并进行成绩记录和排行的六级背单词程序，主要有学习和测验两种功能。

首先打开程序后我们会有一个进入界面，用户可以选择注册或登录。用户有三种可能的身份——管理员、学生、老师。每位用户选择自己合适的身份进入程序进行相关操作。（1）管理员可以进行数据管理；（2）学生可以进行单词学习和单词测试。选择学习时，程序会将题库中随机选择的单词以单个单词形式出现，随机以选择题的形式进行显示，然后等待学生输入自己的答案，如果答案错误，即提示正确答案，如果答案正确，会给予鼓励提示。对于测验板块，程序会给出10道题让学生进行解答，然后计算成绩并记录。第三个大板块就是排行榜系统，程序会采用学生自己与自己比较，还有自己与别的学生进行比较两种方法方便学生了解自己的水平。此外，为了使学生保持高度的学习热情，程序还有设有商店和贪吃蛇功能。学生可以通过测试获得货币，积累到一定数量时，可以选择去商店购买Pikaqiu或者选择玩一次贪吃蛇游戏。（3）老师可以查看学生排行榜，并可以对学生的学习成果进行评论。

为了和其他的单词学习程序与众不同，我们从中挑了几个点精心设计，希望能开创出我们自己的风格。我们的第一个亮点就是在用户注册时会采用容错性机制，即限定用户输入用户名、密码还有邮箱等的格式，比如邮箱就必须有@且后面跟一个合适的域名。第二个亮点就是在进行测验的时候，我们会采用计时工具，这样在之后排行榜上，如果同别人的总分数一样，可以根据所用时间快慢进行划分。第三个亮点就是我们的程序在以学生为主体的同时，还增添了老师这个角色，老师可以对学生的成绩进行评价，给学生一些专业性的建议。第四个亮点是我们程序实现了学习与娱乐相结合，可以帮助学生适当缓解学习压力。

## 分析

我们的项目目的是利用C语言程序设计一个可以让用户进行单词学习的平台，此外也通过排行榜、金钱奖励、老师评价等方式增强学生的学习兴趣，提高学生的学习效率与效果。为了更好地实现这个平台的功能，我们首先创建了三种身份，分别为管理员、老师和学生，之后的操作都是针对每个身份而言的，所以有很多比较繁琐的细节，我们希望以下图的形式展现可以帮助老师更加清晰地理解。

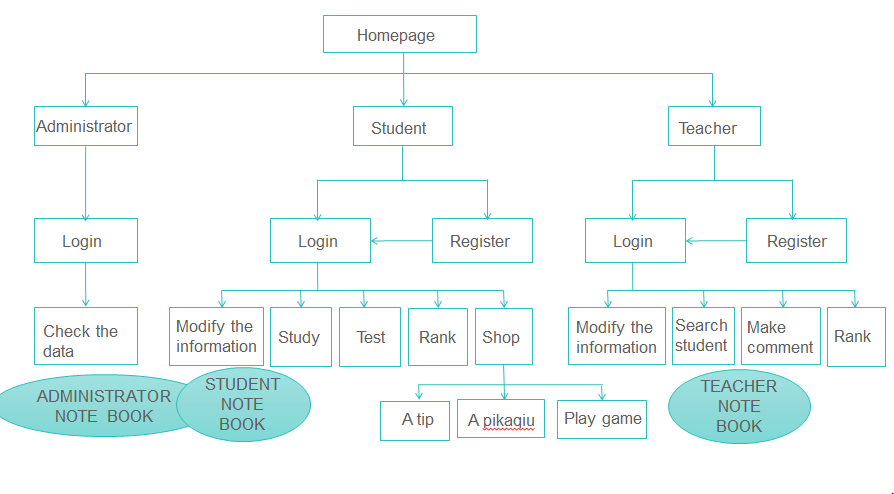


图1-设计流程图

1）信息储存基础

如上图所示，为了令各个用户可以实现登录注册功能，我们选用了记事本来储存他们的信息，并给三个记事本分别命名为addata、tedata和stdata。记事本addata中的信息最简单，仅有用户名和密码两种，中间以逗号隔开，且管理员只支持登陆功能，无法进行注册；记事本tedata中除去用户名和密码，还包含了老师的姓名、性别、电话及所教授的班级；而学生的记事本stdata中所包括信息最多，还有学生的金钱、成绩、对于每道题的熟悉程度等等，之后会依次提到。三种身份的注册与登录功能的具体实现是通过register和login两个函数实现的（每种身份具体函数名不同，管理员没有login函数）。我们的第四个记事本prdata与登录注册无关，是用于存储单词问题的，格式与其他类似，题目为英文单词，四个选项为四个中文词语，其中只有一个是英文单词的正确释义，中间以空格相隔，用户通过中文词语的选择可以进行答题。

2）学生身份功能实现

当以学生身份登录进入系统以后，学生可以进行基本信息的修改，即重新书写记事本并保存，这里的修改项有姓名、ID、密码、电话号码、性别、班级等等。

此外学生还可以进行学习和测试两种操作。学习和测试功能主干部分都是随机从题库中抽取题目并令学生进行作答，然后判断其对错，这两种功能的不同之处主要体现在两个方面：一是我们在学习功能中设计了一个变量degree，并利用它根据学生每次答题正确情况记录特定学生对于每道问题的掌握程度，当掌握程度达到最高等级，即答对次数足够多时，该问题不会再出现。此项功能可以大幅节省学生的时间，提高学习效率。第二处不同在于测试功能中每次做完题目都会有成绩的计算，并与自己之前的最好成绩进行比较，判断是否需要替换成最优成绩，如若进行替换，则该成绩也会实时更新进入排行榜登记于该用户名下，即排行榜是实时更新的（通过函数UpdateRank实现）。成绩的计算有两种情况，我们首先选取分数高的成绩作为最优，但若前后两次分数相同，则取做题所用时间最少的一次作为最优成绩计入。我们相信这样的排行榜模式可以让不同的用户之间形成竞争模式，激励学生们更勤奋努力地进行单词学习。



图2-随机抽取题目并作答示意图

除此以外，我们还设置了“金钱”变量和商店StGoShop函数以实现购物的功能，学生每次完成考试都可以获得一定数目的金钱，可以凭此金钱去商店购买道具。具体可购买道具的设置有三种，分别为“小提示”、“皮卡丘”和“小游戏”。其中小提示可以在用户下一次进行考试或学习时使用，使用一次可以通过循环判断排除一个错误选项，大幅提高学生的答题正确率；皮卡丘是一个可爱的图案收集项，是我们利用软件ASCII Generator 2设计的（此外很多其他的主页面也是我们利用此软件进行设计）；小游戏指的是一次网页版贪吃蛇游戏的试玩，玩家死亡即终止。我们相信这样的商店功能可以激发学生们答题的兴趣，增进他们学习的效率和热情。



图3-收集皮卡丘图案示意图

3）老师身份功能实现

当以老师身份进入系统以后，与学生类似，老师可以进行修改信息和查看排行榜的操作，同时，老师也可以通过函数TeSearch进行查找自己的学生，这里的查找是根据学生的班级是否与老师所对应的班级相符来进行查找的，便于老师查看自己的学生的成绩、排名等信息，更好地进行针对性教学，也大大提高了教学的效率和效果。老师的另一个功能即对学生进行评价，此功能的函数名称为TeMakeComment。这里需要先输入学生的ID，进行搜索以后才可以进行评论。具体的评论内容是通过选项的形式呈现的，老师可以从四个选项“Brilliant”、“Good”、“Okay”、“Ewwwww”中选择其一，该选项所对应的单词会写入学生的记事本，并在学生下一次查看排行榜时呈现，便于查看（若无评价则不显示）。老师的评价可以进行反复评价或进行修改。这项功能可以帮助老师对学生进行指导，学生也可以通过老师的评价获得鼓励或者刺激，激发学生的学习动力。

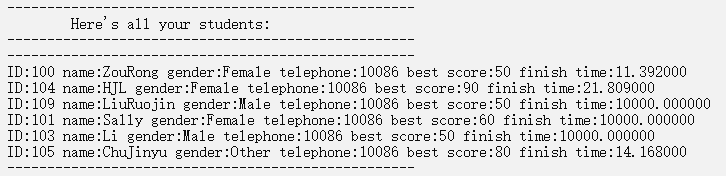


图4-老师搜索学生信息示意图

4）管理员身份功能实现

当以管理员的身份进入时，用户只有一个功能可以实现，即查看所有储存有信息的记事本，包括储存老师、学生信息和记录单词的记事本。为此我们采用TeacherData、StudentData和ProblemData三个函数实现。管理员用户只可以进行登录，不可以注册。我们设置它的目的是便于更好地在软件内进行信息数据的管理。

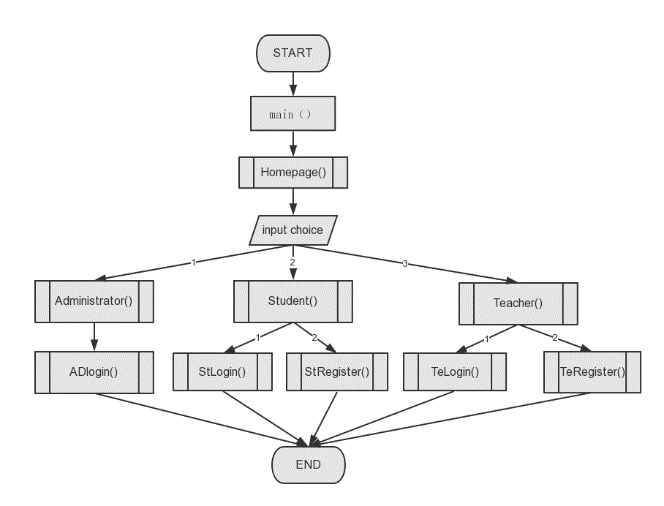
我们所设计软件的中心功能是令学生们学习单词，其他的细节都是为了帮助学生们提高学习效率，有更好的学习体验等等。目前市面上这类的单词学习软件数不胜数，我们只凭借C语言这一项工具，也只是试图将其细节做到最完美、用户体验最完善的基础上尽量进行创新，比如增加商店功能等等，希望能独创出属于自己的风格。

## 设计

1）信息储存的基础

我们所设计的背单词系统设计许多数据的存储，所以在进入功能性函数代码设计之前，我们需要建立相应的记事本文件和相应的结构体数组对不同用户的基本信息进行填写、存储、更改、更新。

2）登入系统部分

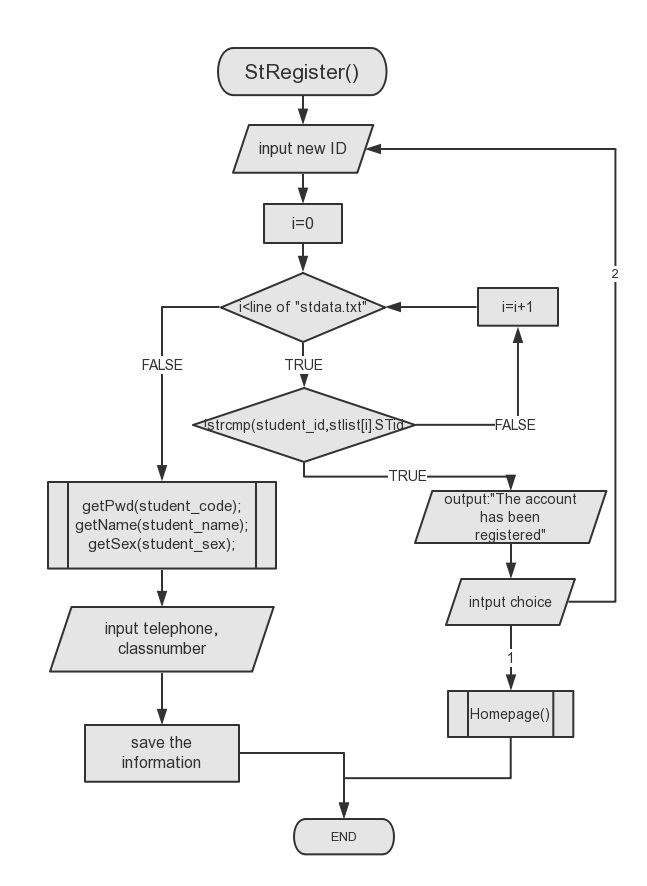
首先系统需要一个主界面，这可通过简单的打开txt文件的函数来实现。在进入主界面后，用户需要选择身份。我们设计的背单词系统所面向的用户分为三种身份，分别是管理员、学生和老师。因此，在进入背单词系统的核心功能使用界面之前，我们可以通过设计相应的选择结构来实现用户的身份选择过程，达到不同身份进入不同界面从而执行不同功能的目的。

3）用户注册部分

用户选择身份之后，需要注册一个新账号才能进入系统进行相应操作。首先应提示用户输入他想要注册的新账号，而系统内每个学生用户都应该是唯一的，所以我们需要避免账号被重复注册的情况，即系统账号应有唯一性。因此，我们需要判断该账号是否已经被老用户注册，为实现这个判断过程，我们将设计相应代码，把所输入的新账号与储存该系统所有已注册学生用户信息的记事本内的账号信息进行比较，如果学生信息记事本内存在学生账号与该新账号一致的情况，说明此账号已被使用，则新用户不能利用这个账号成功进行注册。系统需要提示用户重新输入另一个新账号，并重复上述过程，直至输入的账号是未被使用过的，此时用户就可用此账号进行身份注册。

再确认输入的账号是未被注册的账号后，用户即可进行注册的下一步操作，即进行基本信息的设置，包括登录密码、姓名、性别、班级和电话号码等基本信息。在用户将基本信息输入后，系统将获取输入的信息，并将用户的账号和其他基本信息写入储存学生信息的记事本内。这样一个新用户就成功注册了。

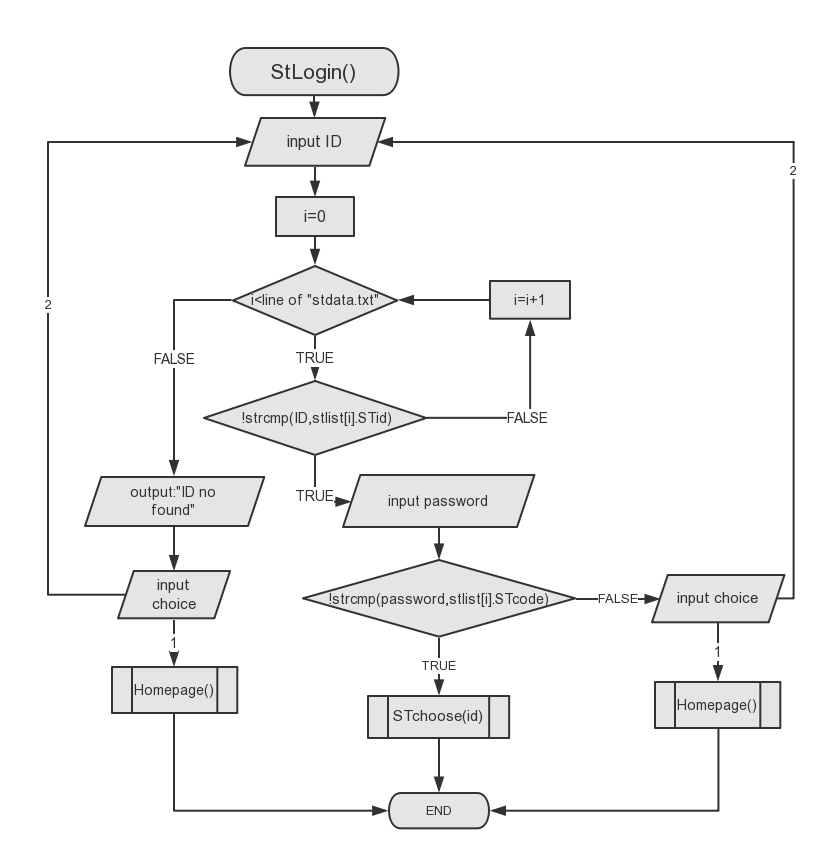
除了学生用户外，教师用户的注册也是类似的，因此，不再作另外的介绍。



4）用户登录部分

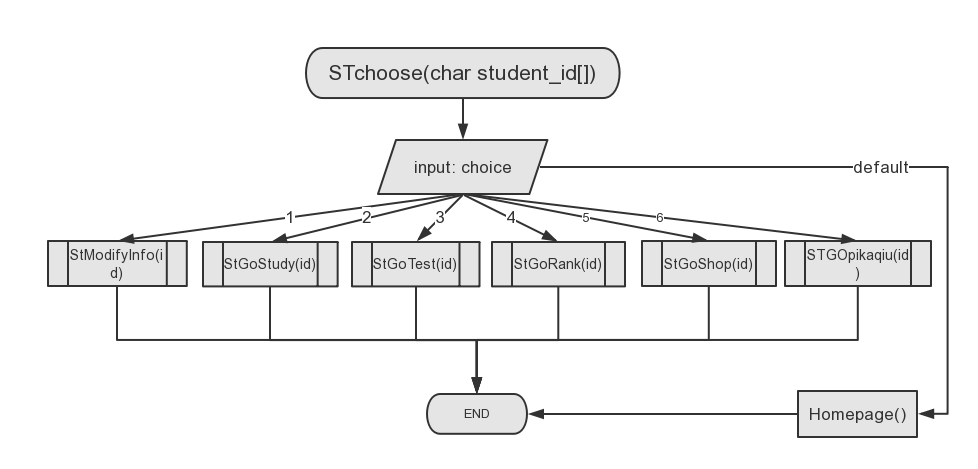
学生用户有了属于自己的系统账号后，就可以用该账号登录背单词系统。登录的过程总体上可分为两个步骤。

第一个步骤为用户输入账号，此时系统需要获取用户输入的账号，并试图在学生信息记事本中找到该账号及该账号的用户相关信息，而这个过程中存在用户输入错误的账号的可能，即系统无法在学生信息记事本中找到该账号，此时需要提示用户输入错误并重新输入账号，并重复上述过程，直至输入的账号是已被注册过的。在用户输入了已被注册的账号后，系统需要在记事本中 读取该账号已设定的密码。

之后就进行第二个步骤，即用户输入密码，系统判断用户是否输入与该账号匹配的正确密码。此时系统需要获取用户键入的信息，并把它与上一步中从学生信息记事本里读取的该账号密码信心进行比较，若两者一致，即用户正确输入了之前设定的登录密码，用户就可以顺利；若这两者不一致，即用户输入了错误的登录密码，系统需要提醒用户输入的密码错误并需要重新输入，用户可再次输入密码，并且系统重复上述过程，直至用户输入了正确的密码，就可以成功登录到系统，进行下一步操作。

除了学生用户之外，其他身份的用户登录，即管理员用户登录、教师用户登录的设计都是类似的，因此，不再作另外的介绍。

5）学生用户操作选择部分

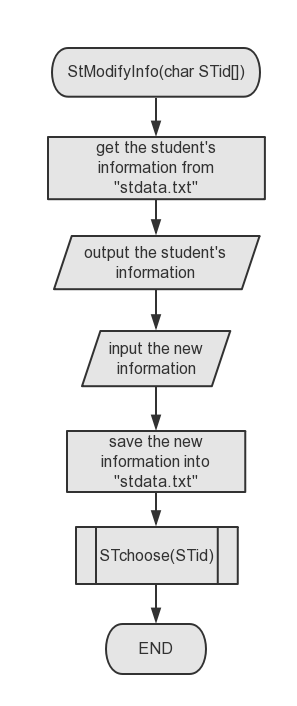
学生用户成功进行登录后，就可以进入背单词系统的功能部分，进行他的下一步操作。我们设计的背单词系统中，学生用户主要可以进行六种操作，分别是修改用户基本信息、单词学习、单词测试、查看排行榜、去商店购买物品、查看自己从商店购买的皮卡丘，这六种功能我们都需要分别写一个函数从而实现其功能。而在用户的选择操作部分，我们可以通过选择结构来实现，根据用户不同的选择，进入不同的函数，从而进行不同功能的操作。

6）用户修改基本信息部分

学生用户可进行其中一个主要操作即为修改自己的基本用户信息。这个过程实际上即为更改学生信息记事本特定位置信息的过程。

首先，系统需要确认记事本中需要修改信息的具体位置，即根据用户账号在记事本中逐行查找对比，直至找到该账号，系统就读取该账号的所有基本信息，包括登录密码、姓名、性别、班级和电话号码等。

之后，系统提示用户输入要更改的信息，而系统便获取用户键入的信息，并将其保存到记事本中相应的位置中，这就实现了更改基本信息的目的。

除了学生用户之外，其他身份的用户基本信息修改，都是类似的，因此，不再作另外的介绍。

7）学生用户进行单词测试部分

单词测试功能主要可分为以下几个过程：

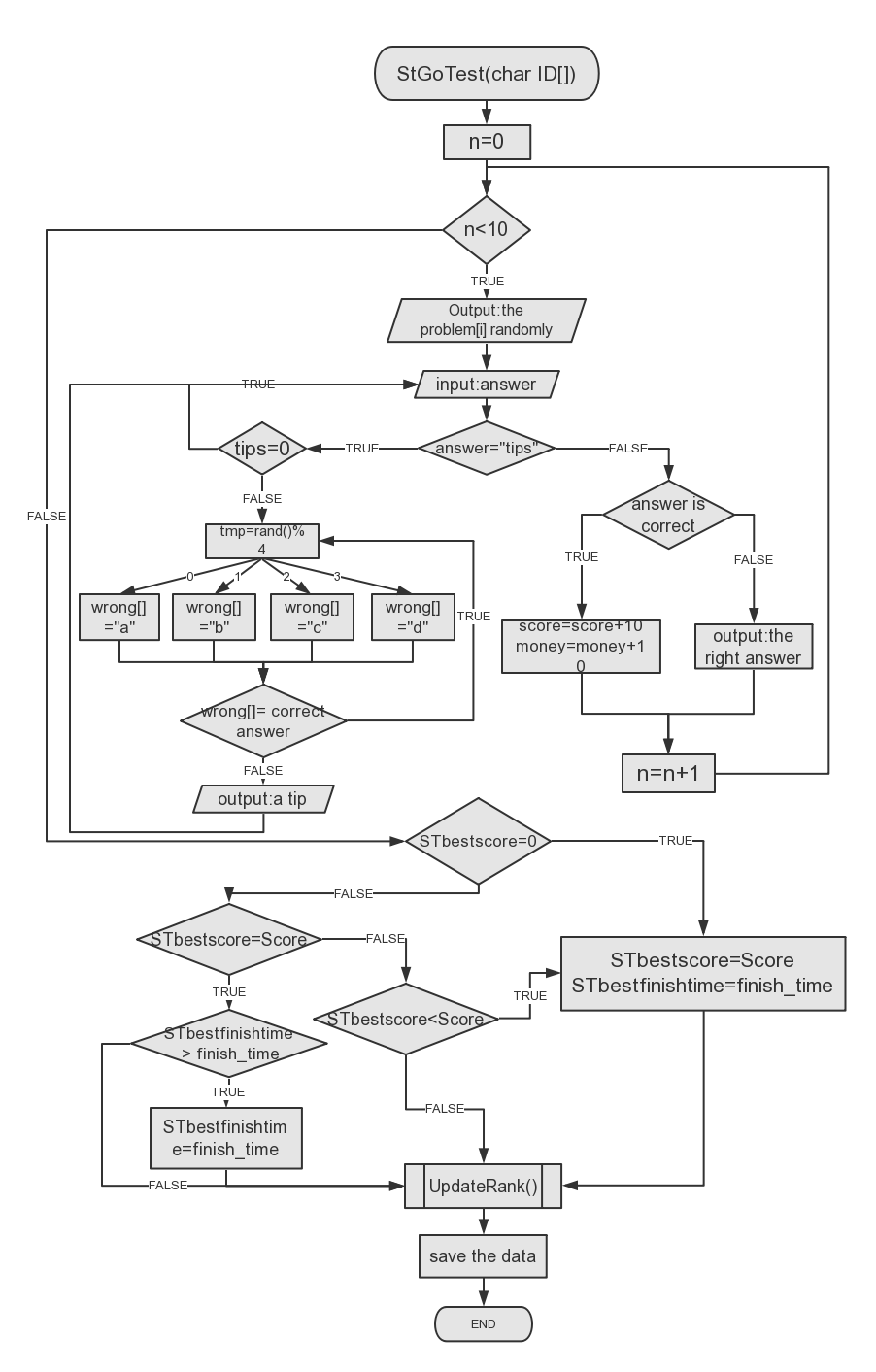
首先，系统从单词库记事本中随机读取试题的问题与正确答案信息，并将问题部分输出到系统界面，让学生用户可以在系统界面上看到试题。

其次，学生进行答案判断并向系统输入自己的回答。系统需要获取学生键入的信息，将其与上一步从学生信息记事本中读取的正确答案信息进行比较，若两者一致，则学生回答正确，学生的分数增加10分，且学生可以获得相应的“钱”；若两者不一致，则学生回答错误，，系统给出正确答案。在完成这一题的回答与答案判断后，系统给出下一道题，学生继续作答。

如果用户输入了单词“tips”，说明用户认为这道题有难度并希望系统能给他一个提示，以利于他能够正确答题。在背单词系统中，“tips”是需要学生从商店中购买的，而学生所拥有的“tips”的个数储存在学生信息记事本中，因此在系统给出提示之前，需要判断学生拥有的“tips”的个数是否大于或等于1。如果学生用户拥有的“tips”的个数为0，说明他此时没有使用“tips”的机会；如果学生用户拥有的“tips”的个数大于或等于1，则学生可以顺利使用提示功能，并且学生拥有的“tips”的个数要减去一个，此时系统将随机从除了正确选项之外的其他三个选项中随机抽取一个出来，提示用户该选项是错误的，从而使得用户在回答问题时可以排除这个选项提高正确回答的可能性。

在进行了所设定的相应数量的单词测试试题后，学生即完成此次单词测试，系统得出此次测试所得的分数和所用时间，并将其与储存在学生信息记事本中的最好成绩进行对比。若此次成绩低于历史最高成绩，则不进行信息更改。若此次成绩高于历史成绩，则将此次测试的最终成绩与所使用的时间记录到学生信息记事本中。若此次成绩等于历史成绩，则需再判断此时所用的测试时间是否少于以前，如果是则覆盖之前的信息，否则不更改。

最后，将新的测试信息与排行榜中其他学生的成绩进行比较，这里通过上课时所学的排序算法即可实现，因此在此处不做另外介绍。

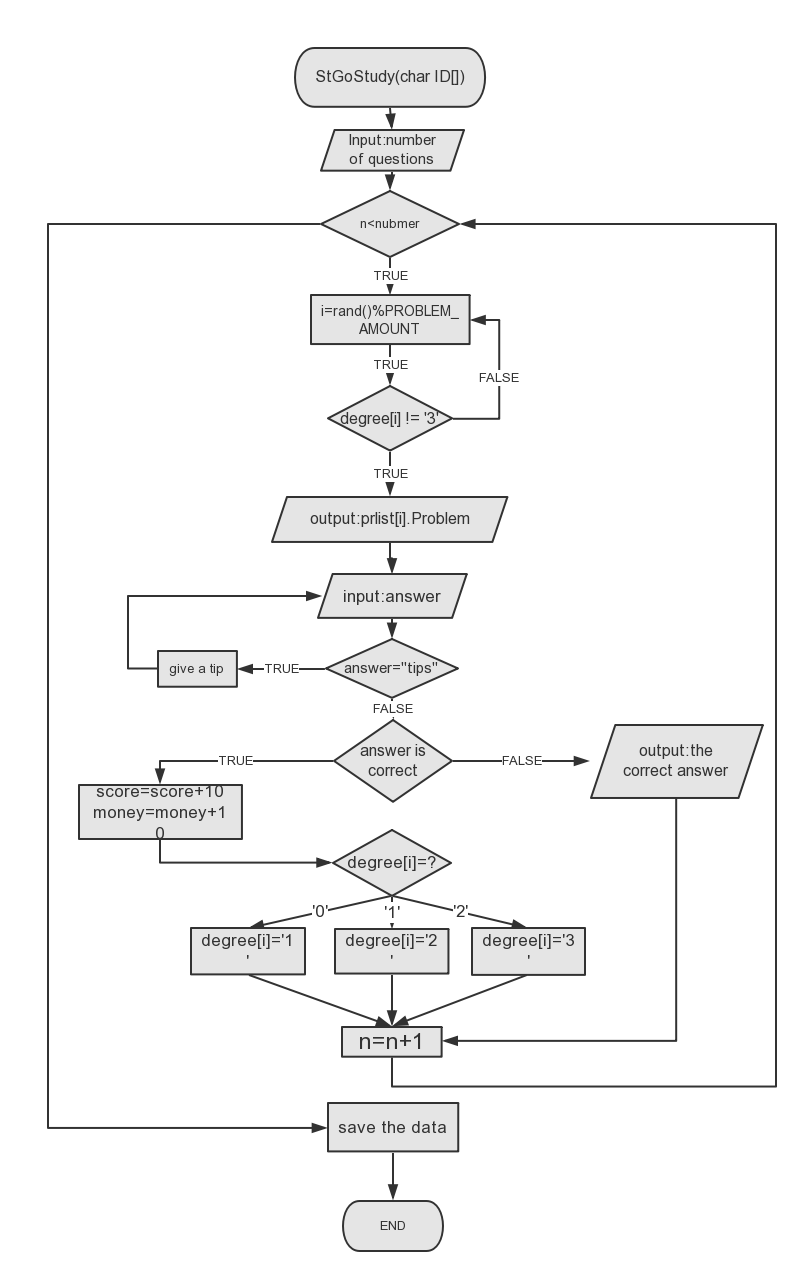


8）学生用户进行单词学习部分

在学习功能中，我们的一个突出点就是degree变量。Degree变量适用于衡量学生对于该单词熟悉程度的变量，每个学生对于每一道问题都有一个degree变量。Degree的可取值为0-3，分别对应含义如下：

|  |  |
| --- | --- |
| Degree 等级 | 含义 |
| Degree 0 | 从未学习此单词 |
| Degree 1 | 成功学习此单词1次 |
| Degree 2 | 成功学习此单词2次 |
| Degree 3 | 成功学习此单词3次，该单词不会再出现 |

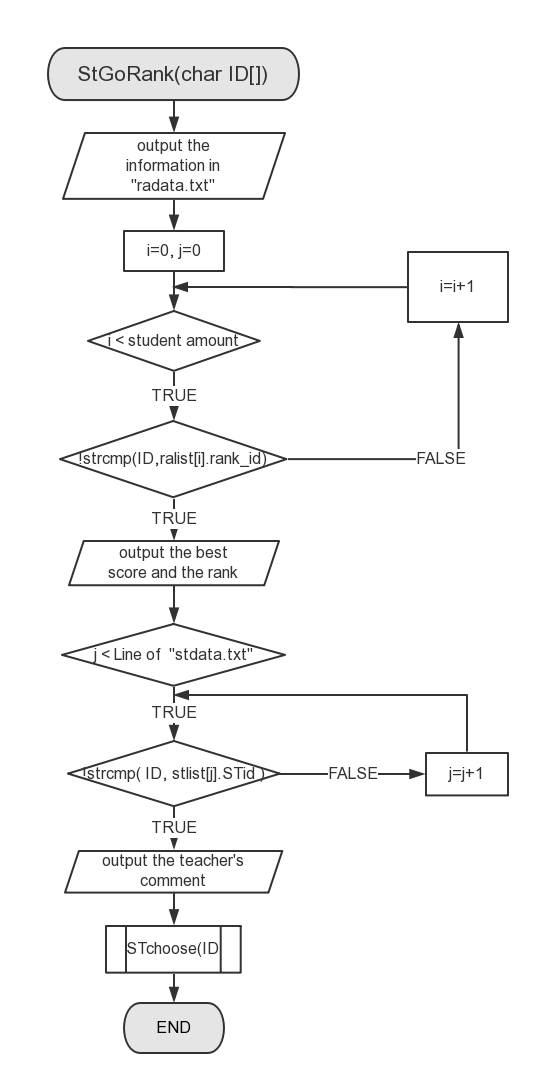
我们利用degree变量根据学生每次答题正确情况记录特定学生对于每道问题的掌握程度，当掌握程度达到最高等级，即等级3时，此时学生答对该问题次数足够多时，该问题不会再出现，可以大幅度提高学生学习效率。



9）学生查看排行榜部分

学生用户可以查看排行榜从而获得自己与其他优秀同学的成绩与排名。而我们在此前就设计创建一个记事本专门用于存储学生的成绩信息与排名信息，所以在查看排行榜功能这个部分中，只要简单读取排行榜记事本中的所有信息并将其输出即可达到查看排行榜的目的。

而在该部分中，除了查看排行榜，学生还可以查看到老师给他的评语。这个部分系统需要根据该学生的账号，在学生信息记事本中查找到该学生的信息，并读取储存在这的老师评语信息，之后进行输出，即可实现这个功能。

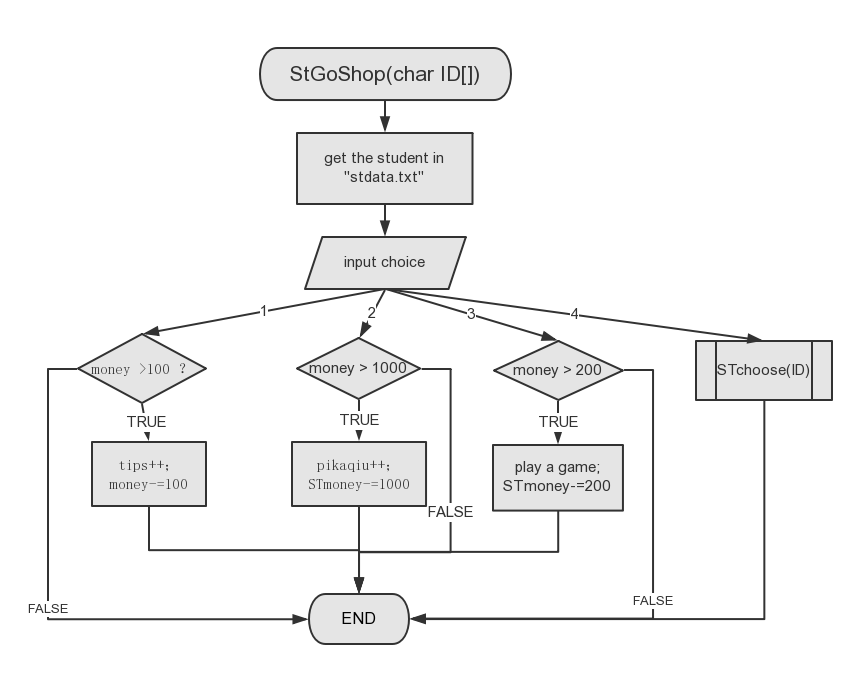


10）商店购物部分

学生在进行了单词学习和测试后，都会得到相应的“钱”，学习与测试的次数越多，获得的“钱”越多，而得到的“钱”就是可以用于到商店购买相应物品的。

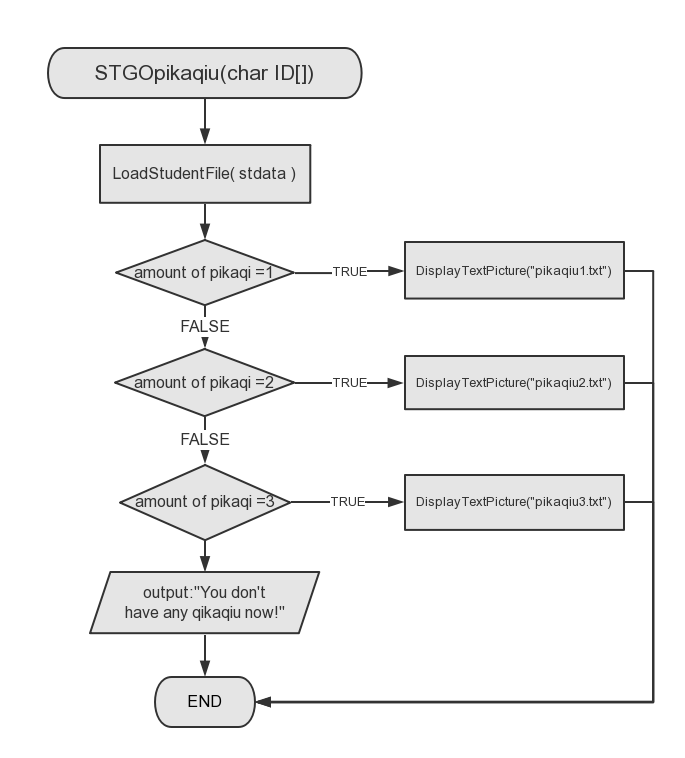
在用户选择购买某种商品后，系统需要判断用户是否拥有足够的“钱”，用户拥有的“钱”的数量存储在学生信息记事本中。若“钱”不够则提示用户不能购买，若足够，用户即可成功购买，且学生拥有的“钱”要减少相应数量。

商店中购买的物品有三种：第一种是可用于单词测试与学习中获取提示的“tips”，在购买“tips”成功后，系统需要将存储在学生信息记事本中的“tips”个数增加一个。第二种商品是卡通角色皮卡丘。第三种商品是一次玩游戏的机会，我们在网上找到了一个贪吃蛇的html格式的文件，因此这个商品成功购买之后，通过相关函数打开该文件即可进行游戏。



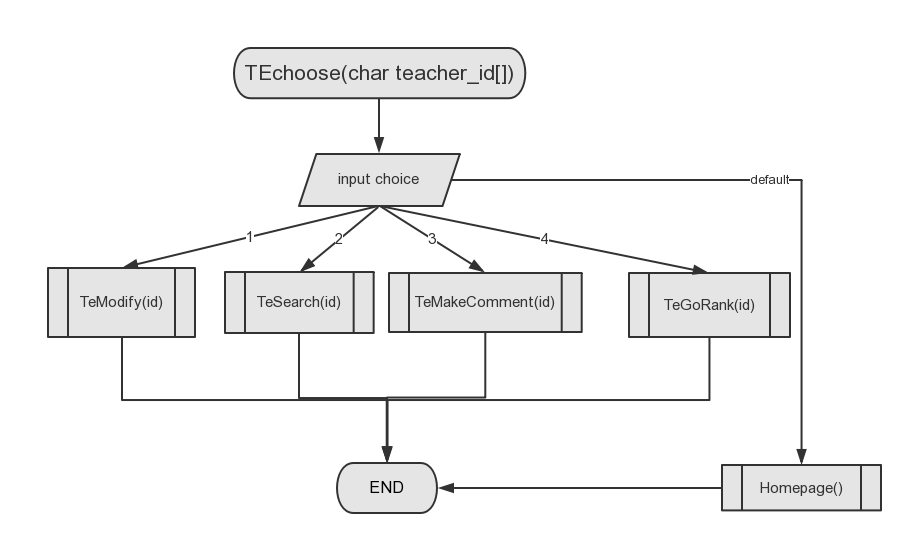
11）查看拥有的皮卡丘

这个部分我们通过简单的相关文件的打开来达到查看皮卡丘的目的。我们在网上找到皮卡丘相关的图片并将其做成txt格式的文件，不同数量的皮卡丘对应不同的文件。若用户购买了一只皮卡丘，则打开第一个文件；若用户购买了两只皮卡丘，则打开第二个文件；若用户购买了三只皮卡丘，则打开第三个文件。

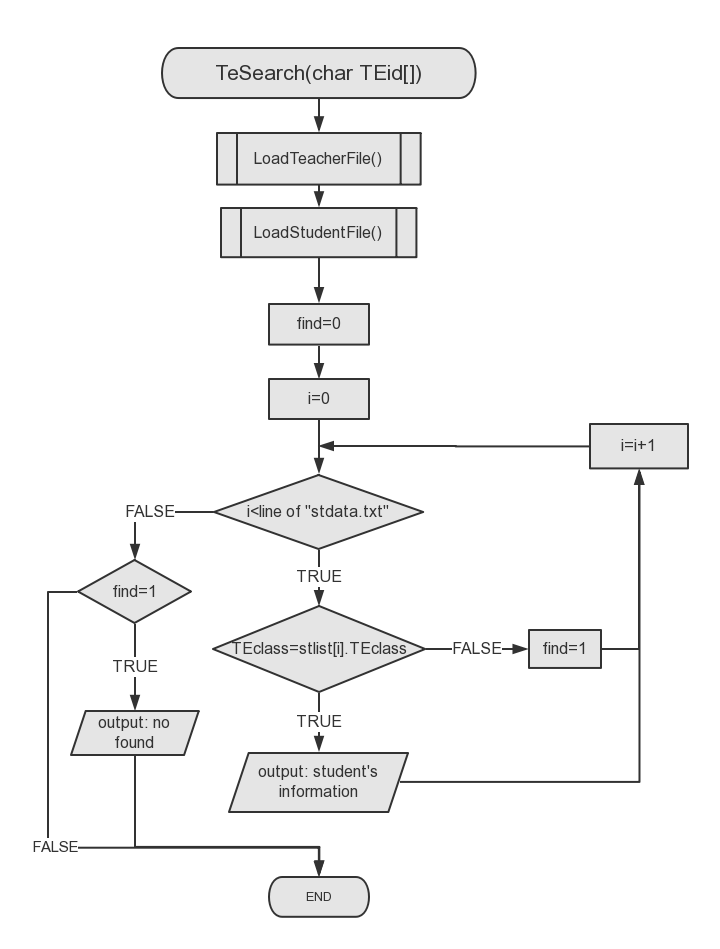


12）老师功能主界面

为了使得界面更加清晰简洁，我们选用TEchoose来进行老师主界面选择。老师可以选择修改信息、搜索学生、评价或者看排行榜。每个选项都通过特定的函数导向对应功能。

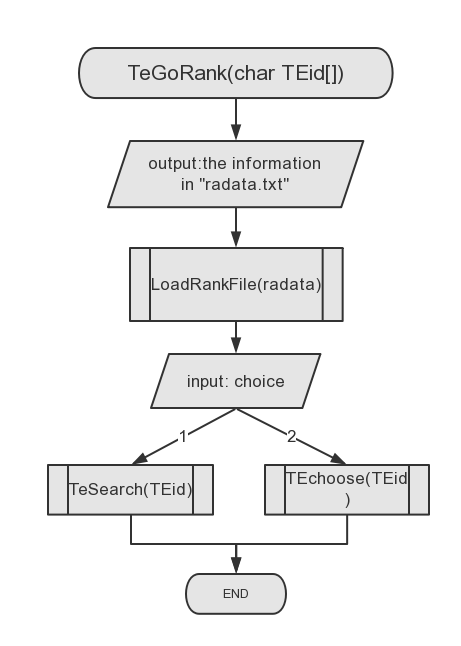


13）老师对于本班学生的查找

由于进行学生信息查找的时候需要老师和学生班级相符合，故首先需要载入老师和学生的记事本并进行信息的读取。在读取到老师的班级后，我们通过循环依次判断学生的班级是否与老师相同，如果相同，即认定此学生为本班学生并将其挑出，在之后显示他的信息，如ID、姓名、性别、电话号码、最佳测验分数和所用时间等等。如果不相同，则需要继续循环，查看下一个学生的信息是否对应。如果直到最后一位学生也未搜到相符合班级的学生，系统会提示“No Found”，并令用户在“重新输入”和“返回主界面”中任选其一。

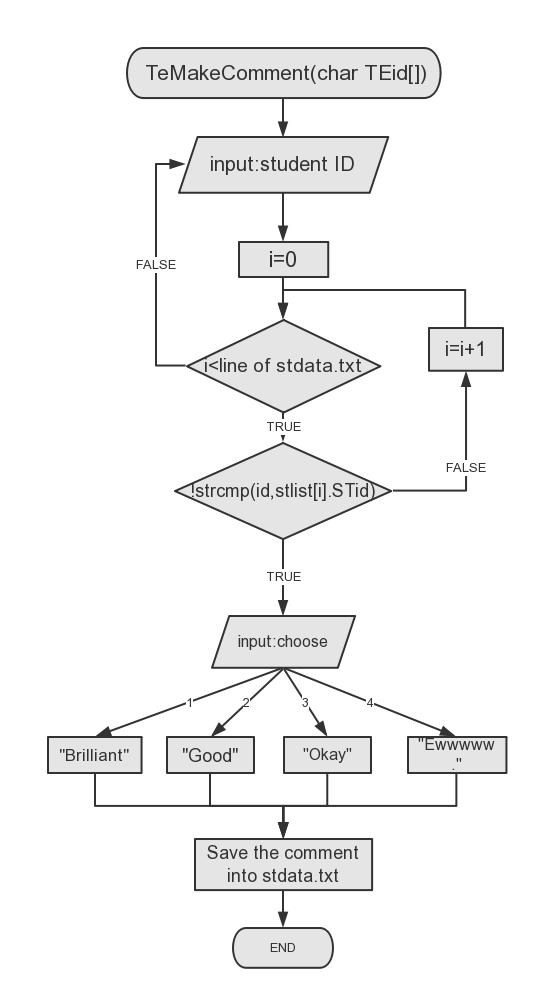
14）老师查看学生的排行榜

老师可以查看学生的排行榜，从而起到监督督促的作用。老师所见排行榜与学生的排行榜思路相同，均是通过比较来实现，故不做过多讨论。不同点在于，老师最后可以选择是否需要仅查看自己的学生。如果老师要求查看，则会导入上一个函数TeSearch中，其中有学生们的最好成绩及所用时间，便于老师对于本班学生完成考试的情况进行检查。



15）老师对学生作出评价

在排行榜的最初，老师需要输入欲评价学生的ID，以便进行针对此学生进行评价。我们采用函数通过循环查看是否有符合标准的ID存在，若有则挑出，若未找到会提示“该ID未进行注册，不可使用”。当找到了对应学生后，老师可以在四个评语中选择一个并进行图案的修饰。该评论会被记录于排行榜中，当所对应的学生下一次查看排行榜时即可见。

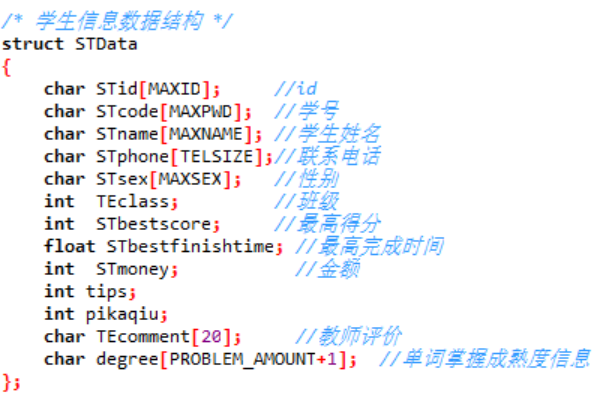


## 实现

我们使用Dev-C++来进行代码的编写、调试与实现。

1）信息储存

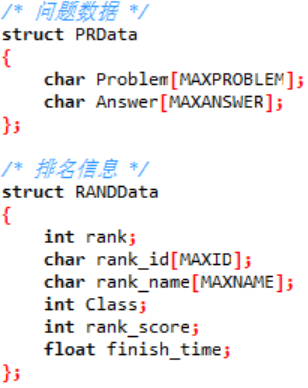
①用户信息的储存

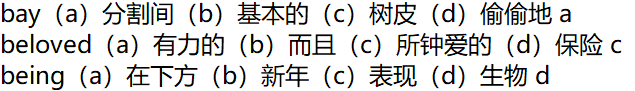


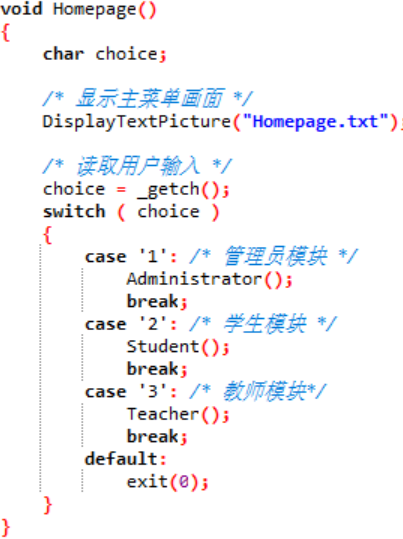
以学生用户的信息储存为例，我们建立了一个学生信息的结构体数组STdata，在该数组里面，包括了账号、登录密码、姓名、性别、班级、电话号码、最高得分、最好完成时间、金额、“tips”个数、皮卡丘个数、教师评价、单词掌握成熟度总共13个元素。之后，在注册时会将这些数据写入到相应的记事本中，记事本中某一行储存了某一个学生的信息。

②题库的储存学生信息

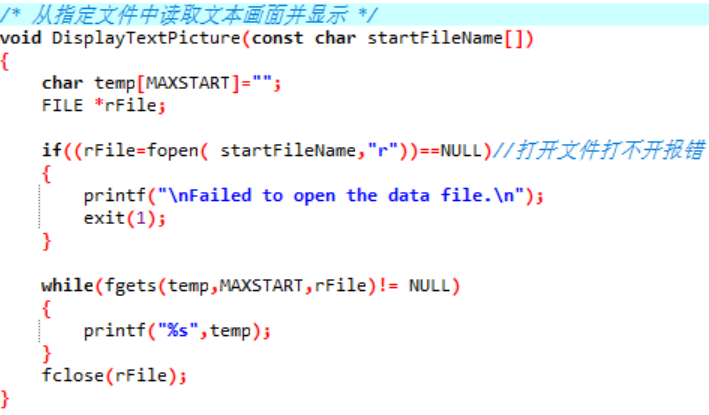
对于题库，我们建立了一个结构体数组PRData，它包含了问题与正确答案两个元素。我们人工将试题输入到题库记事本prdata.txt中，且在问题与正确答案之间需要输入一个空格，以便程序读取时能够区分问题与答案。



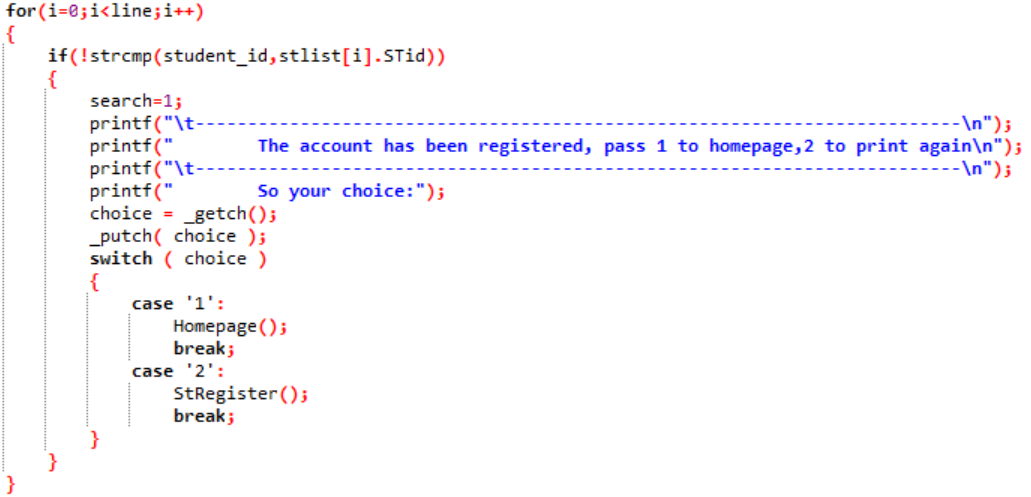


2）登入系统部分

我们的主界面的显示采用打开txt文件的方式实现，因为在多个函数中都需要打开txt作为面面，所以我们定义了一个DisplayTextPicture(const char startFileName[])函数以简化我们的程序代码。在这个函数中我们用if语句考虑了可能会出现打不开对应文件的情况，避免程序崩溃。



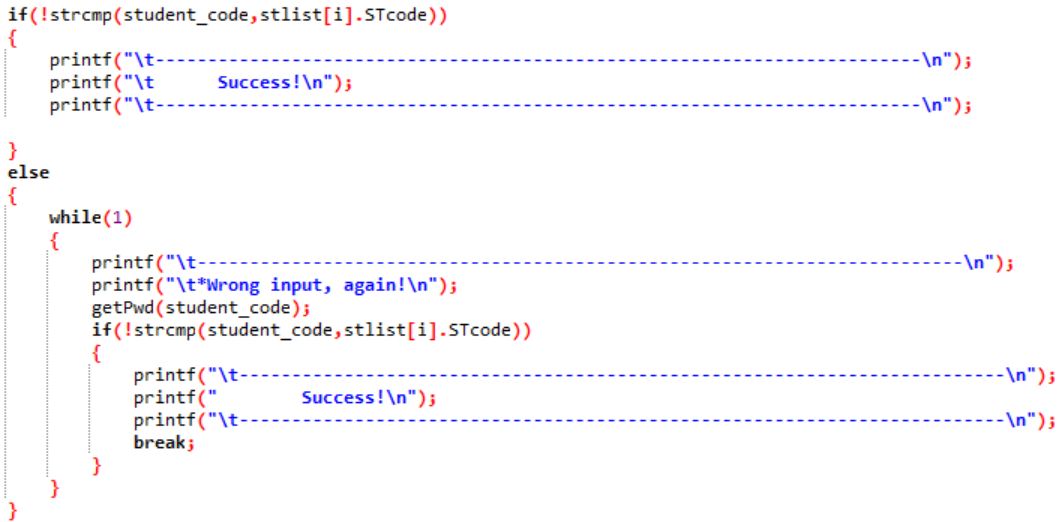
3）用户注册部分



在注册部分，我们采用循环语句for，将记事本中的老用户账号逐个与新用户输入的账号进行比较，再用if语句和strcmp()函数来判断该用户想要注册的账号是否已经被注册。

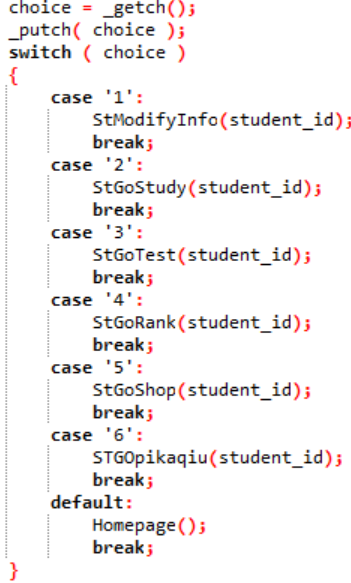
C:\Users\黄嘉铃\Desktop\注册写入.png注册写入在用户顺利注册并填写了他的基本信息之后，我们用fprintf()函数将用户输入的数据信息写入到学生信息记事本stdata.txt中，保存新用户信息。

4）用户登录部分

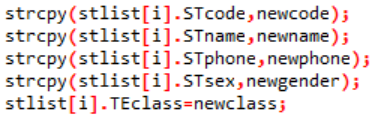
登录部分的关键就是判断用户是否输入正确的密码，如图中，student\_code即为用户输入的登录密码，而stlist[i].STcode即为从学生信息记事本stdata.txt中读取的学生设置的密码，再用if()和strcmp()函数比较两个字符串是否一致，若一致则登录成功，若不一致则进入while()循环语句，要重新输入直至输入正确的密码之后才跳出循环结构。

5）学生用户操作选择部分

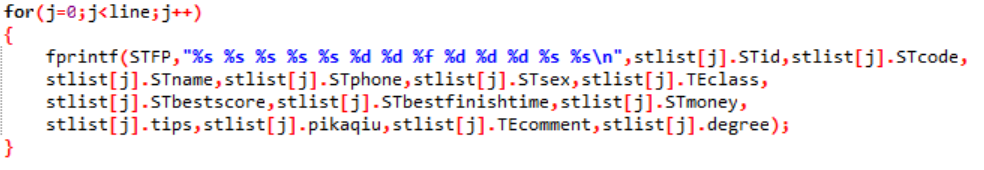
在这部分我们使用\_getch()读取用户输入的信息，即用户的操作选择，再用switch语句根据不同的选择进入不同的功能界面。



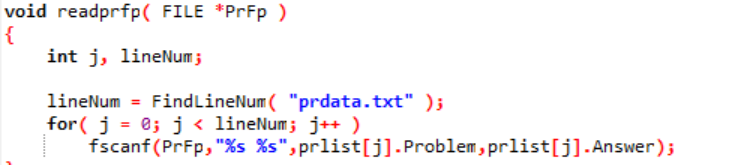
6）用户修改基本信息部分



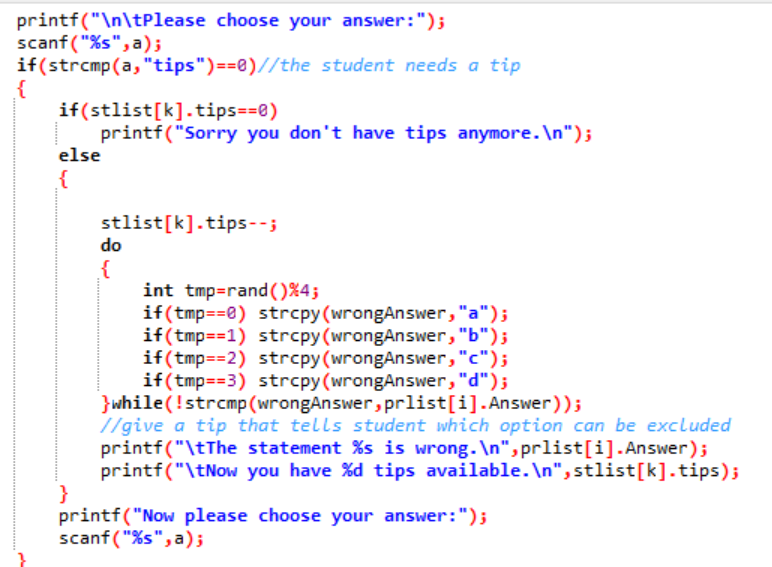
在这部分，主要采用一个strcpy()将用户输入的新信息赋值到相应结构体数组的元素中。

之后，再使用fprintf()将新数据写入到记事本中，更新用户信息。

7）学生用户进行单词测试部分

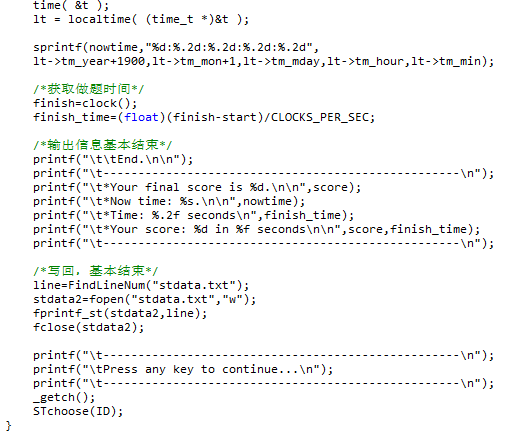
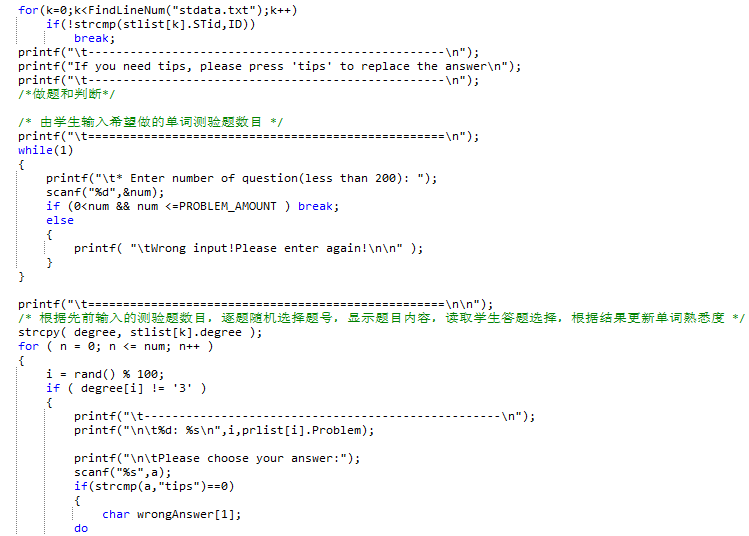
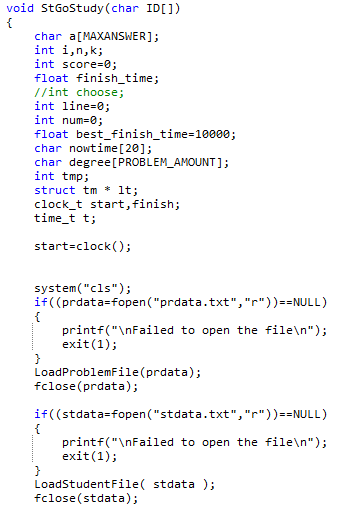


首先，在进行单词测试前，系统需要对词库信息进行读取，因为有多个函数都需要对词库信息进行读取，所以我们定义了readprfp(FILE \*PrFp)函数用于读取信息。



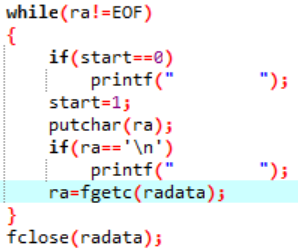
之后对于单词测试的“tips”功能，我们首先strcmp()判断用户是否输入了“tips”，如果是就进入“tips”部分。用if()判断用户是否有足够的“tips”。用rand()%4从a、b、c、d四个选项中随机抽取一个选项，而提示功能是用于告知用户某一错误选项可以排除，所以随机抽取出来的选项必须是错误选项，所以我们用strcmp()确保抽取出的选项字符不可以与题库中设置的正确选项字符一致，若一致则进去do...while循环结构，直到能够抽取出错误选项才跳出循环。

8）学生用户进行单词学习部分

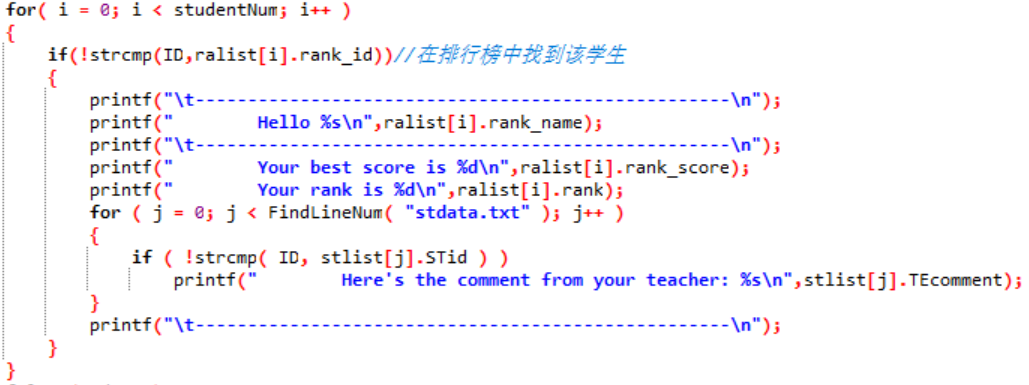


学生学习界面与测试界面的最大区别在于degree变量的设定，所以我们也将主要讲述这一部分的代码。在学生注册功能时，我们即使用menset函数将注册用户对于每道题的degree值设置为0，以便之后进行熟悉度增加。

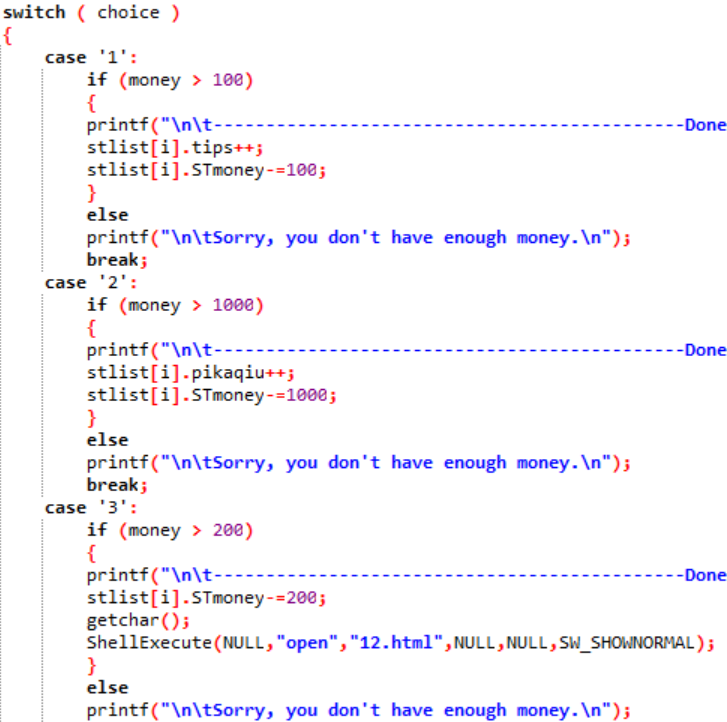
首先，在随机选择完题目以后，我们通过if函数来判断学生对于该题目的degree值是否为3，如若为3，则需重新选择题目，以保证学生只需要回答degree小于3的题目。之后，因为degree是字符型，所以在学生答对题目以后，选择使用switch函数对于degree值进行增加，如若学生答题错误，则没有该步骤，degree值保持不变。最后，通过strcpy将degree信息写回记事本。

9）学生查看排行榜部分

首先，我们用while循环结构和putchar()、fgetc()对排行榜记事本radata.txt中的字符逐个读取、逐个输出，直至读取到文件的结尾，就实现了输出排行榜所有数据的目的。

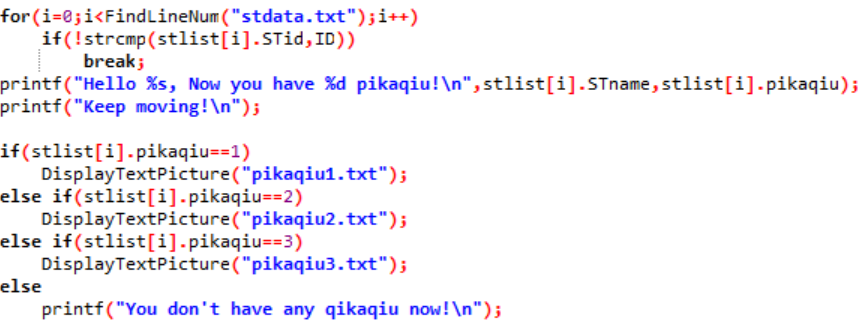
 之后，用for循环、if语句、strcmp()函数在排行榜记事本中找到该学生对应的排行榜信息，并且再用一个for循环在学生信息记事本中找到该学生并输出老师给他的评语信息。

10）商店购物部分



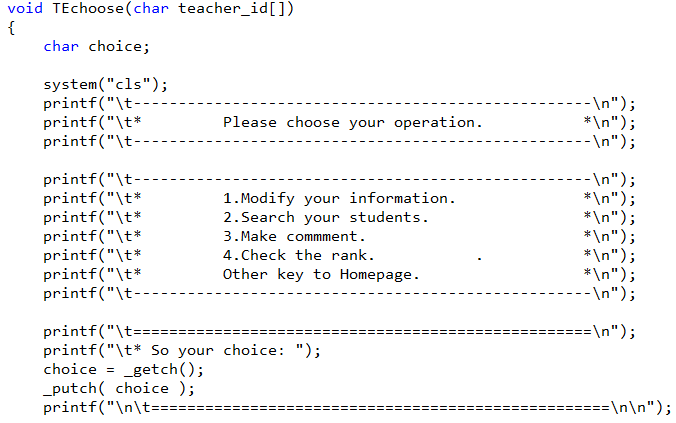
读取用户输入的数据后，利用switch语句判断用户想要选择购买什么商品，并用if语句判断用户是否有足够的购买该商品的金钱。在第三个商品，即购买一次玩游戏的机会中，我们用ShellExecute (NULL,"open", "12.html",NULL, NULL,SW\_SHOWNORMAL)函数直接打开贪吃蛇游戏的html文件，12.html即为该游戏文件。

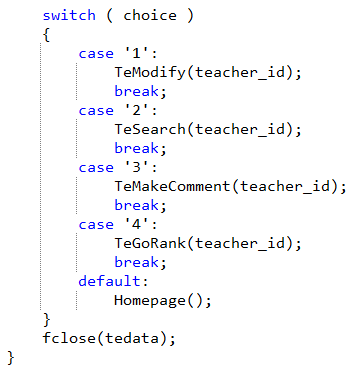
11）查看拥有的皮卡丘

在显示用户所拥有的皮卡丘之前，用for循环、if和strcmp()在学生信息记事本中找到该用户并读取该用户所拥有的皮卡丘数量信息，即学生结构体数组中的stlist[i].pikaqiu元素的数值。之后根据这个元素的数值，打开相应的皮卡丘图片txt格式的文件，如pikaqiu1.txt中只有一只皮卡丘，pikaqiu2.txt中有两只皮卡丘。

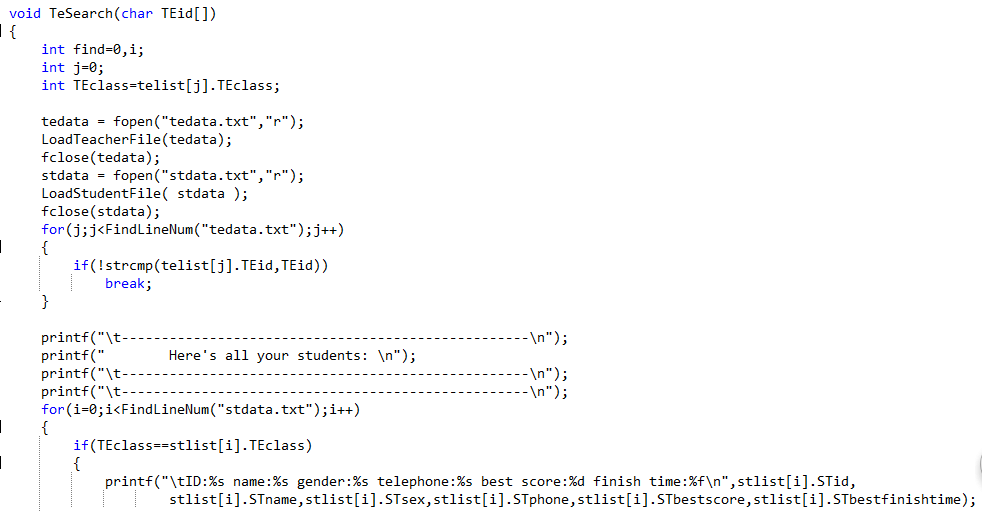
1. 老师功能主界面

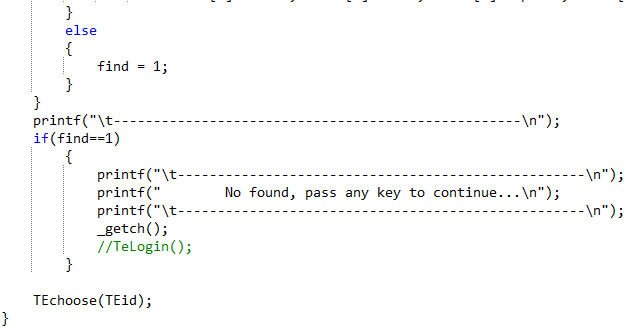
这部分我们使用\_getch()读取用户输入的信息，即用户的操作选择，再用switch语句根据不同的选择进入不同的功能界面。





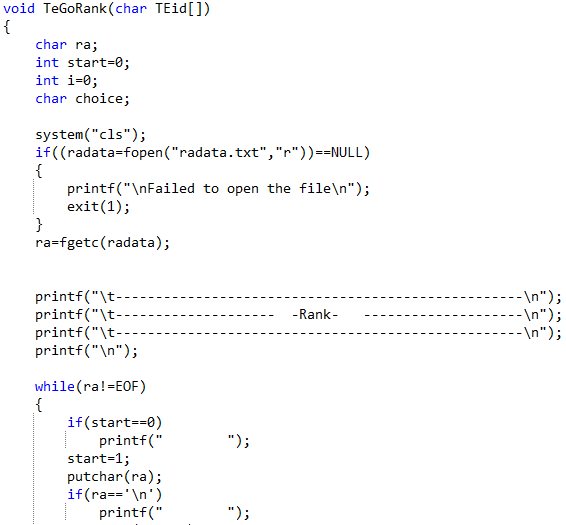
13）老师对于本班学生的查找

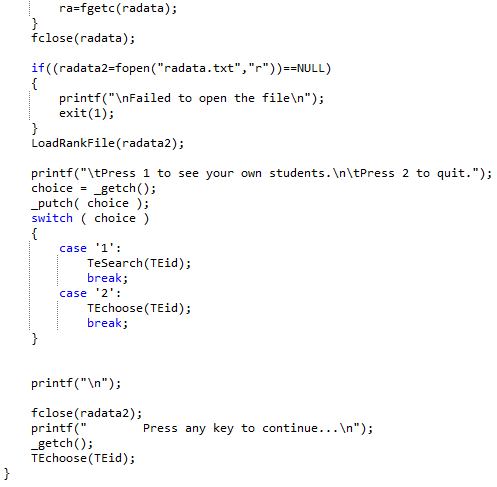




在老师查找学生界面，我们首先通过fopen和fclose对老师、学生储存信息的记事本信息进行读取，然后通过for循环遍历学生信息，通过if语句依次判断学生的班级是否与老师班级对应，如若对应，则通过printf将学生信息展示出来，遍历到信息列表结尾终止。如若未找到任何对应学生，则通过printf显示“No Found”以提示老师，并指引进行下一步操作。

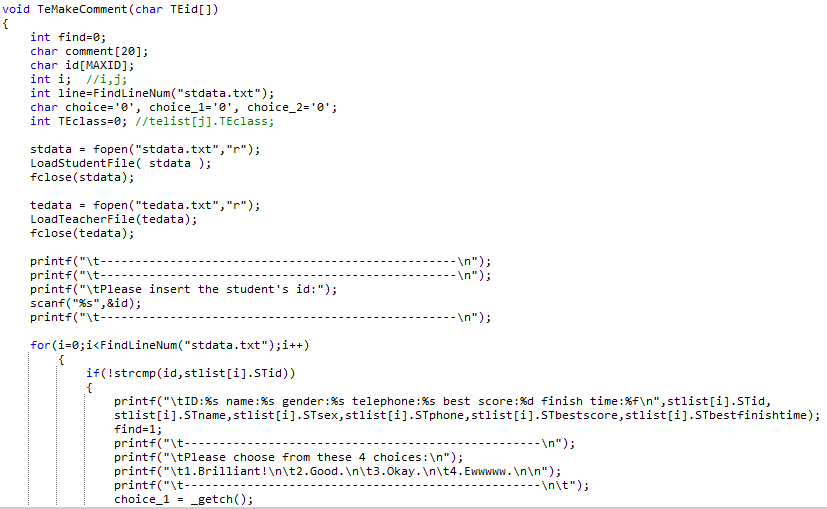
14）老师查看学生的排行榜



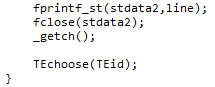


老师查看排行榜的代码与学生类似，我们首先用while循环结构和putchar（）、fgetc（）对排行榜记事本radata.txt中的字符逐个读取、逐个输出，直至读取到文件的结尾，就实现了输出排行榜所有数据的目的。

15）老师对学生作出评价







在老师对学生进行评价的阶段，我们首先通过scanf指引用户输入欲评价学生的ID，之后与查找学生类似，通过for循环遍历，查找到相应学生并通过switch函数选择四个评价中一个，反之，若未找到符合ID的学生，则会提示“该ID未注册”。

## 调试与测试

|  |  |  |  |
| --- | --- | --- | --- |
|  | 输入 | 预想结果 | 真实结果 |
| 主界面 | 输入’a’ | 不能运行程序 | 程序自动退出 |
| 注册 | 输入一个已被注册过的账号 | 不能注册 | 此账号不能注册，并提示用户“The account has been registered.” |
| 输入一个未被注册过的账号 | 能注册 | 能注册，并提示用户输入密码 |
| 输入一个长度超过11位字符的账号 | 不能注册 | 不能注册，并提示用户账号过长，重新输入 |
| 输入新密码时输入了长度超过19个字符的字符串 | 不能设置成该密码 | 不能设置成该密码，并提示用户超出长度重新输入 |
| 登录 | 输入一个用户信息记事本内不存在的账号 | 不能登录 | 不能登录，并提示“ID no found, press 1 to homepage,2 to print again” |
| 输入一个用户信息记事本内存在的账号 | 能继续操作 | 能继续操作，并提示用户输入密码 |
| 输入错误的密码 | 不能登录 | 不能登录，并提示”Wrong input, again!” |
| 输入正确的密码 | 成功登录 | 成功登录 |
| 学生进行单词学习 | 在提示学生输入想要学习的单词数量时输入’r’ | 不能运行 | 程序出现错误，显示”Wrong input” |
| 在提示学生输入想要学习的单词数量时输入’5’ | 进行5个单词的习题学习 | 进行5个单词的习题学习 |
| 管理员操作页面 | 输入“1”选择查看学生信息 | 显示系统中所有学生用户的信息 | 显示系统中所有学生用户的信息 |
| 输入“2”选择查看教师信息 | 显示系统中所有教师用户的信息 | 显示系统中所有教师用户的信息 |
| 输入“3”选择查看题库信息 | 显示系统中题库的试题与正确答案 | 显示系统中题库的试题与正确答案 |
| 输入“r” | 返回到主界面 | 返回到主界面 |
| 学生操作页面 | 输入“1”选择更改信息 | 进入更改信息界面 | 进入更改信息界面 |
| 输入“2”选择进行学习 | 进入单词学习界面 | 进入单词学习界面 |
| 输入“3”选择进行测试 | 进入单词测试界面 | 进入单词测试界面 |
| 输入“4”选择查看排行榜 | 显示完整排行榜 | 显示完整排行榜 |
| 输入“5”选择去商店 | 进入商店界面 | 进入商店界面 |
| 输入“6”选择查看皮卡丘 | 显示拥有的皮卡丘 | 显示拥有的皮卡丘 |
| 输入不是选项的字符“r” | 不能进行进一步操作 | 程序自动返回到主界面 |
| 学生进行单词测试 | 输入”tips” | 给出提示 | 给出提示，如 ”The statement a is wrong.” |
| 输入“1” | 显示回答错误 | 给出提示 “Wrong! The right answer is %s.”并显示下一题 |
| 在a、b、c、d中输入一个正确答案 | 回答正确 | 回答正确，，并显示下一题 |
| 在a、b、c、d中输入一个错误答案 | 回答错误 | 回答错误，提示 ”Wrong! The right answer is %s.”并显示下一题 |
| 学生去商店购买商品 | 在钱不够的时候选择购买一个商品 | 不能成功购买 | 不能购买，并提示“Sorry, you don't have enough money.” |
| 在钱足够的时候选择“3”购买一次玩游戏的机会 | 可正常购买并进行游戏 | 可正常购买，并弹出游戏游戏界面，开始游戏 |
| 输入不是选项的字符“e” | 不能购买 | 不能购买并提示“Please press any key to go back to choose your operation.” |
| 输入“4”选择“I don't want to buy anything。” | 返回上一个页面 | 返回上一个页面 |
| 学生更改信息 | 输入新密码时输入了长度超过19个字符的字符串 | 不能更改 | 不能更改成该密码，并提示用户超出长度重新输入 |
| 将密码更改为“122222” | 更改成功 | 更改成功，提示“Change successfully” |
| 老师查找学生 | 记事本中有一个学生与老师班级对应 | 输出该学生信息 | 输出学生信息（姓名、班级、最好成绩等） |
| 记事本中有没有学生与老师班级对应 | 无法输出 | 提示“No Found”，并指引老师回到功能选择界面 |
| 老师给学生评论 | 已有学生ID | 输出信息，等待评论 | 输出学生信息，并提示老师可以进行评论 |
| 不存在学生ID | 无法输出，无法评论 | 提示“It's an unregistered ID”，并提示老师重新输入 |
| 老师操作页面 | 输入“1”选择更改信息 | 进入更改信息界面 | 进入更改信息界面 |
| 输入“2”选择进行老师查找学生 | 进入老师查找学生界面 | 进入老师查找学生界面 |
| 输入“3”选择老师评论学生 | 进入老师评论界面 | 进入老师评论界面 |
| 输入“4”选择查看排行榜 | 显示完整排行榜 | 显示完整排行榜 |
| 输入不是选项的字符“r” | 不能进行进一步操作 | 程序自动返回到主界面 |
| 老师更改信息 | 输入新密码时输入了长度超过19个字符的字符串 | 不能更改 | 不能更改成该密码，并提示用户超出长度重新输入 |
| 将密码更改为“122222” | 更改成功 | 更改成功，提示“Change successfully” |
| 修改性别为”male” | 更改成功 | 更改成功，提示“Change successfully” |
| 修改ID为“3532” | 更改成功 | 更改成功，提示“Change successfully” |
| 修改姓名为“hhh” | 更改成功 | 更改成功，提示“Change successfully” |
| 修改班级为“17b” | 更改成功 | 更改成功，提示“Change successfully” |
| 修改电话号码为“1324324413” | 更改成功 | 更改成功，提示“Change successfully” |

## 结果与结论

经过一学期的时间，我们基本实现了六级背单词程序最初的设计目的。管理员可以直接进行登录，然后进行相关数据管理。学生可以进行单词的学习和测试、查看排行榜和老师评论、进行一些娱乐活动缓解学习压力。老师可以查看学生排行榜并进行评价。我们在整个项目的实现过程中不断地发现问题并解决问题，并改变了一些初衷。一开始我们计划的是程序里面会设计不同的班级，每个班级的老师只能对自己班的学生进行评价，但由于自身能力问题，没有让程序实现这个功能，目前此程序里老师可以对每个班学生的学习成果进行评价，这样恰好也能让学生获得更多的专业性建议。

这次六级背单词程序的设计过程，使我们受益匪浅。在词库的录入过程，顺便加深了我们自身对单词的记忆力。整个程序的建立过程不仅提升了我们运用C语言知识的能力，更是训练了我们的思维——如何去发现问题、解决问题。