

**程 序 设 计 专 题**

**大 程 序 报 告**



1. **学生姓名 ： 李家豪 学号： 3180104460**
2. **学生姓名 ： 朱亿江 学号： 3180104457**
3. **学生姓名 ： 朱佳鹏 学号： 3180104456**

**2018~2019春夏学期 2019 年 6 月 10 日**

# 题目描述和题目要求

基于libgraphics，设计一个图形应用程序，对学生信息进行录入，管理。要求能够实现以下几点：

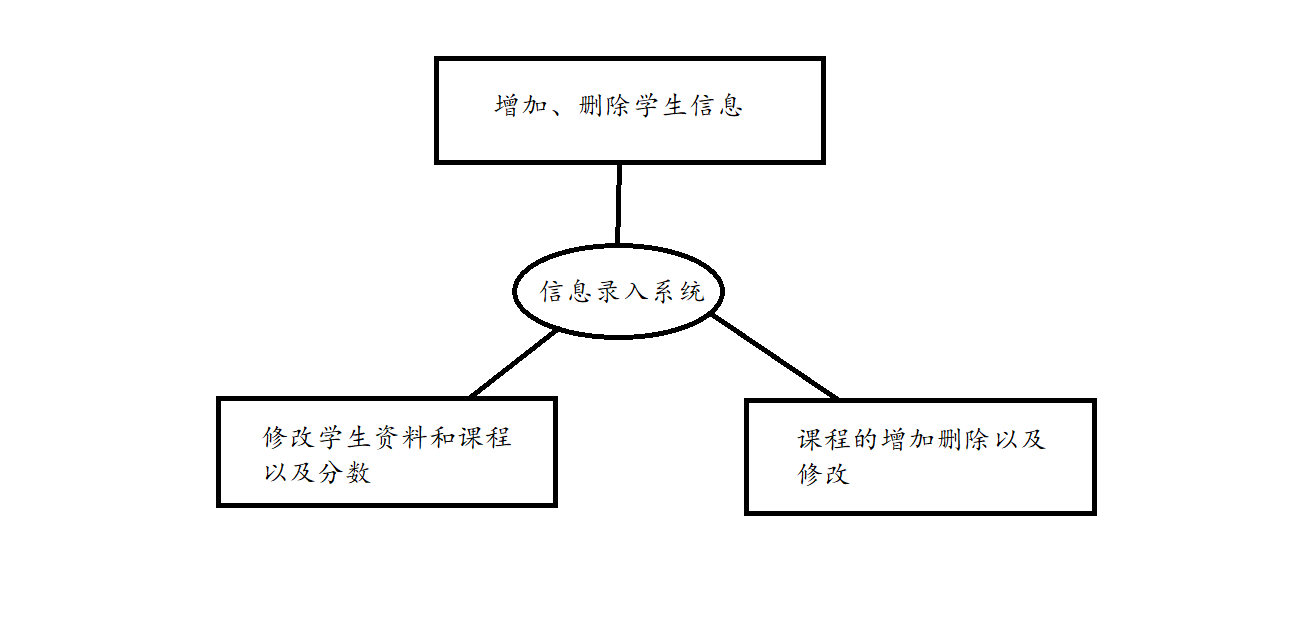
1. 文件能够执行新建、打开、保存、关闭、退出等等操作
2. 学生和课程信息能够执行新增、复制、粘贴、删除等等操作
3. 有“帮助”模块，能够解释本软件的使用方法、注意事项等等
4. 有“菜单”模块，其中包含常用功能，不少于5个
5. 在窗口底部即时显示操作的中间状态，结果等等
6. 能够实现建立和修改学生和课程的关系
7. 学生条目至少包括：姓名、学号、专业、课程列表
8. 课程至少包括：编号、名称、选课名单和成绩情况
9. 能够实现学生信息和课程情况之间的相互导向
10. 根据学生和课程或其他信息内容，设计多种统计计算并展示结果
11. 按照姓名或者学号排序，并显示排序结果
12. 设计链表用于系统数据的组织
13. 设计适当的文件格式，保存学生信息和他们的课程信息
14. 系统程序必须采用多文件组织的方式

# 需求分析

本系统要求能实现交互设计，要求能实现以下几要素：菜单系统、图标工具栏、快捷键、状态信息栏、各个功能（编辑功能、检索功能、统计功能、排序功能）、文件读取。

# 3、总体设计

## 3.1 功能模块设计



## 3.2 数据结构设计

|  |  |  |
| --- | --- | --- |
| 数据结构类型 | 声明 | 功能介绍 |
| 链表 | typedef struct Course{  char number[30];  char coursename[30];  int score;  struct Course\* next;  struct Student \*s;  }course; | 变量名为course的链表节点结构变量  Number[30]——课程序号  Coursename[30]——课程名称 |
| typedef struct Student{  char id[30];  char name[30];  char major[30];  char phonenumber[30];  int score;  course \*c;  struct Student \*next;  }student; | 变量名为student的链表节点结构变量  Id[30]——学号  Name[30]——学生姓名  Major[30]——专业名称  Phonenumber——电话号码 |
| 数组 | static char status[80]; | 存储状态栏内容信息 |
| static char filename[80] ; | 存储文件名信息 |
| static char clipboard[80]; | 存储剪贴板信息 |
| static char addid[12]; | 存储新增加的学号信息 |
| static char addname[16]; | 存储新增加的学生姓名信息 |
| static char addmajor[30]; | 存储新增加的专业信息 |
| static char addphonenumber[12]; | 存储新增加的电话号码信息 |
| static char addcourses[12][30]; | 存储新增课程信息的二维数组 |
| static char addscores[12][10]; | 存储新增加的成绩信息 |
| static char addcoursenumber[30]; | 存储新增加的课程序号信息 |
| static char addcoursename[30]; | 存储新增加的课程名称信息 |
| static char search[80]; | 存储搜索信息 |

说明：在学生结点中除了有学生基本信息之外，还有一个该学生所选的课程的链表；在课程结点中除了有课程基本信息之外，还有一个选该课程的所有学生组成的链表。

P.S：其实在数据结构这一部分我们小组设计得还不是很好，现在呈现出来的这种设计比较繁琐，后续进行程序功能的扩展时会极其不方便。更好的做法是，在学生结构体里套一个含有该学生所选的课程的指针的链表，在课程结构体里套一个含有选该课程的所有学生的指针组成的链表，即学生的个人信息和课程的个人信息在内存中都只有一份，而学生选课和查看选课名单等操可以用指针来完成。这样可以保证学生自身信息和课程自身信息的唯一，简化程序操作，提高程序运行速率。但是由于我们在临近提交作业时才想到这一设计，没有足够时间做大面积的代码重写，所以只能先沿用之前的设计。

## 3.3 函数功能描述

void DisplayClear(void);

功能描述：清屏，由libgraphics提供

void DrawBox(double x, double y, double width, double height);

功能描述：画出矩形边框

参数描述： x,y为矩形边框左下角，width,height分别为长宽的值

函数算法描述：先将画笔移到（x，y）处，再用DrawLine（dx，dy）函数画出四条直线

int addtolinkedlist()；

功能描述：将一个新的学生结点加入到学生链表中并修改

返回值描述：若课程不存在则返回0，存在则返回1

函数算法描述：新建一个学生结点并将该学生所有课程加入到学生结点信息中，并在课程链表中加入该学生的信息

void Addtocourse(course \*head, char \*s1, char \*s2, int n)；

功能描述：将一个课程结点加入到课程链表中

参数描述： head为链表头，s1为课程编号，s2为课程名称，n为分数

函数算法描述：将一个课程结点加入到课程链表中

void readtolinkedlist(FILE \*fp)；

功能描述：将信息从文件中获取，并生成链表

参数描述： fp为文件指针

函数算法描述：不断地读取文件，根据已经设定好的文件格式分析文件内容，分隔出字符串，然后已有函数生成学生和课程链表

void writetofile(FILE \*fp)；

功能描述：将信息从链表中提取出来，保存至文件中

参数描述： fp为文件指针

函数算法描述：遍历学生链表，读取学生信息和选课情况，按照已有格式将全部信息写入文件中

void showhelp();

功能描述：显示帮助信息

重要局部变量定义：x为总宽的1/5，y为总长的6/7，h为字体高度的四倍

重要局部变量用途描述：x，y，h用于确定文字框的位置

函数算法描述：确定顶部文字框的位置并输入文字，再依次确定其余文字框的位置并输入。

void showabout();

功能描述：显示关于信息

重要局部变量定义：x为总宽的1/5，y为总长的2/3，h为字体高度的四倍

重要局部变量用途描述：x，y，h用于确定文字框的位置

函数算法描述：确定顶部文字框的位置并输入文字，再依次确定其余文字框的位置并输入。

void showdetails(student \*p);

功能描述：显示学生的详细信息，包括学号、姓名、专业、手机号码及各科成绩。

参数描述：输入某个已经建立的学生链表结点指针

重要局部变量定义：整数i，j，初始值为0。

重要局部变量用途描述：用于列表计数，i最大为6，一列最多6门课程，j最大为12，最多12门课程。

函数算法描述：以列为单位显示学生列表“详细信息”的界面

void guiling();

功能描述：参数归零

函数算法描述：把所有用于判断状态的数值全部调为0，即回归初始状态

void idsort(student \*Head, int reversed);

功能描述：按学号对学生链表进行排序

参数描述：输入某个已经建立的学生链表结点指针，reserved确定是正序还是逆序

函数算法描述：类似冒泡排序，但是不移动数据，只修改每个结点的指针

void namesort(student \*Head, int reversed);

功能描述：按姓名对学生链表进行排序

参数描述： 输入某个已经建立的学生链表结点指针，reserved确定是正序还是逆序

函数算法描述：类似冒泡排序，但是不移动数据，只修改每个结点的指针

int idisexist(char \*s);

功能描述：判断学生学号是否已存在

参数描述：输入一串待判断的字符串

返回值描述：若已存在则返回1，不存在则返回0

函数算法描述：遍历学生链表

int cnumberisexist(char \*s);

功能描述：判断课程编号是否已存在

参数描述：输入一串待判断的字符串

返回值描述：若已存在则返回1，不存在则返回0

函数算法描述：遍历课程链表

int nameisexist(char \*s);

功能描述：判断学生姓名是否已存在

参数描述：输入一串待判断的字符串

返回值描述：若已存在则返回1，不存在则返回0

函数算法描述：遍历学生链表

void searchstudent(char \*s);

功能描述：搜索指定学生

参数描述：输入一串待搜索的字符串

重要局部变量定义：整数i，k，初始值为0

重要局部变量用途描述：i作为遍历的游标，k用于暂存结果的数量

函数算法描述：将搜索栏输入的值与每名学生的学号，姓名，专业，电话号码相比较，如相同则保存这名学生的结点指针到数组中

void studentdelete(student \*p);

功能描述：删除某一学生

参数描述：输入该学生的结点指针

函数算法描述：将此结点从学生链表和所有课程中除去

void coursedelete(course \*s);

功能描述：删除某一课程

参数描述：输入该课程的结点指针

函数算法描述：将此结点从课程链表和所有学生中除去

void destroystudentlist();

功能描述：销毁学生链表

函数算法描述：依次释放学生链表所有结点

void destroycourselist();

功能描述：销毁课程链表

函数算法描述：依次释放课程链表所有结点

void replaceblank(char \*s);

功能描述：把字符串中的空格转换成$符号

参数描述：输入某一字符串

函数算法描述：从首地址开始依次将字符串中的空格转换成$符号

void dereplaceblank(char \*s);

功能描述：把$符号还原为空格

参数描述：输入某一字符串

函数算法描述：从首地址开始依次将字符串中的$符号转换成空格

student\* createstudentnode();

功能描述：新建一个学生结点

返回值描述：返回指向该结点的指针

函数算法描述：malloc()一块空间，并将学生属性初始化

course\* createcoursenode();

功能描述：新建一个课程结点

返回值描述：返回指向该结点的指针

函数算法描述：malloc()一块空间，并将课程属性初始化

void Addtocourselist(student \*s, char \*s1, char \*s2, int score);

功能描述：在读取文件时将学生所选的课程加到课程链表中

参数描述：正在进行操作的该学生结点指针，课程编号s1，课程名称s2和该学生这门课程的分数score

函数算法描述：遍历现有的课程链表，判断该课程是否存在。若存在，则添加该学生信息；若不存在，新建一个该课程结点并添加该学生信息

course\* courseisexist(char \*name);

功能描述：根据课程名称判断该课程是否已存在

参数描述：输入某个课程名称

返回值描述：返回一个指针，若不存在，则为空指针；若存在，则此指针指向课程名称所在结点

函数算法描述：遍历课程链表，将课程名称与链表中的课程名称相比较

void showsearchresult(student \*\*s);

功能描述：展示搜索结果

参数描述：传入一个储存学生结点的指针数组

函数算法描述：依次输出该数组中所有学生的信息

void drawpageselector(int choose);

功能描述：显示翻页器

参数描述：输入翻页器种类

函数算法描述：根据当前页码与最大页码的大小分别进行前翻与后翻

void showxuanke(student \*\*s, char \*number);

功能描述：显示选课名单

参数描述：一个储存学生结点的指针数组和该课程的编号字符串

函数算法描述：将课程编号与每名学生信息中的课程编号相比较并输出结果

course\* searchcourse(char \*s);

功能描述：搜索某一课程

参数描述：输入待搜索的字符串

返回值描述：此课程的结点指针

函数算法描述：遍历课程链表，将字符串与链表中的课程名称与编号相比较

void display(void);

功能描述：显示界面

函数算法描述：先清屏并显示状态栏，再根据当前状态决定显示什么

## 3.4 程序文件结构

1）文件函数结构

add.h add.c(在.h文件中声明，在.c文件中定义，以下略)

void Addtocourselist(student \*s, char \*s1, char \*s2, int score);

int addtolinkedlist();

void Addtocourse(course \*head, char \*s1, char \*s2, int n);

courses.h courses.c

void coursesmenu(int selection);

void showcourseinterface();

void showcourseaddinterface();

void showcoursedeleteinterface();

void showcoursesearchinterface();

void showcoursesearchresultinterface();

void showcoursereviseinterface();

void showcourselist();

void showcoursestatistics();

void showcoursedeletinginterface();

void showcourseinfo();

create.h create.c

student\* createstudentnode();

course\* createcoursenode();

destroy.h destroy.c

void destroystudentlist();

void destroycourselist();

display.h display.c

void display();

void DisplayClear();

void drawMenu();

void drawstatus();

void drawtoolbar();

eventprocess.h eventprocess.c

void CharEventProcess(char ch);

void KeyboardEventProcess(int key, int event);

void MouseEventProcess(int x, int y, int button, int event);

file.h file.c

void filemenu(int selection);

void guiling();

void createfile();

void savefile();

void openfile();

void closefile();

void readtolinkedlist(FILE \*fp);

void writetofile(FILE \*fp);

void drawopeninginterface();

void drawclosinginterface();

void drawsavinginterface();

void drawquittinginterface();

help.h help.c

void showhelp();

void showabout();

void helpmenu(int selection);

void drawhelpinginterface();

void drawaboutinginterface();

others.h others.c

int cnumberisexist(char \*s);

int idisexist(char \*s);

int nameisexist(char \*s);

void searchstudent(char \*s);

void coursedelete(course \*s);

void studentdelete(student \*s);

void replaceblank(char \*s);

void dereplaceblank(char \*s);

course\* courseisexist(char \*name);

void showsearchresult(student \*\*s);

void showxuanke(student \*\*s, char \*number);

course\* searchcourse(char \*s);

void readytosearch();

pageselector.h pageselector.c

void drawpageselector(int choose);

void changepageselector(int choose);

showtable.h showtable.c

void showaddcoloum(double x, double y, int choose);

void showcoursebox(int i, course \*p);

void showcourses(course \*p, int choose);

void showstudentbox(int i, student \*p, char \*score);

void showstudents(student \*p, int choose);

void DrawBox(double x, double y, double width, double height);

void idsort(student \*Head, int reversed);

void namesort(student \*Head, int reversed);

students.h students.c

void studentsmenu(int selection);

void showstudentinterface();

void showstudentaddinterface();

void showstudentsearchinterface();

void showstudentsearchresultinterface();

void showstudentreviseinterface();

void showstudentdeleteinterface();

void showstudentdeletinginterface();

void showcoloums();

void showdetails(student \*p);

Main.c

void Main()

includes.h

对基础库的引用

global.h global.c

对全局变量的声明及定义

2）多文件构成机制

（所有头文件均加入了#pragma once，是条件编译，防止同一头文件被重复包含）

add.h

#include "includes.h"

#include "global.h"

#include "others.h"

#include "create.h"

courses.h

#include "global.h"

#include "includes.h"

#include "others.h"

#include "pageselector.h"

#include "create.h"

#include "showtable.h"

#include "add.h"

create.h

#include "includes.h"

#include "global.h"

destroy.h

#include "global.h"

#include "includes.h"

display.h

#include "includes.h"

#include "global.h"

#include "file.h"

#include "students.h"

#include "courses.h"

#include "help.h"

eventprocess.h

#include "global.h"

#include "includes.h"

#include "display.h"

file.h

#include "includes.h"

#include "global.h"

#include "destroy.h"

#include "showtable.h"

#include "add.h"

#include "others.h"

#include "create.h"

help.h

#include "global.h"

#include "includes.h"

#include "file.h"

others.h

#include "includes.h"

#include "global.h"

#include "showtable.h"

pageselector.h

#include "includes.h"

#include "global.h"

showtable.h

#include "includes.h"

#include "global.h"

#include "pageselector.h"

#include "others.h"

students.h

#include "includes.h"

#include "global.h"

#include "pageselector.h"

#include "others.h"

#include "showtable.h"

#include "add.h"

#include "create.h"

includes.h

对基础库的引用

global.h

#include "includes.h"

Main.c

#include "global.h"

#include "includes.h"

#include "eventprocess.h"

# 4.部署与运行

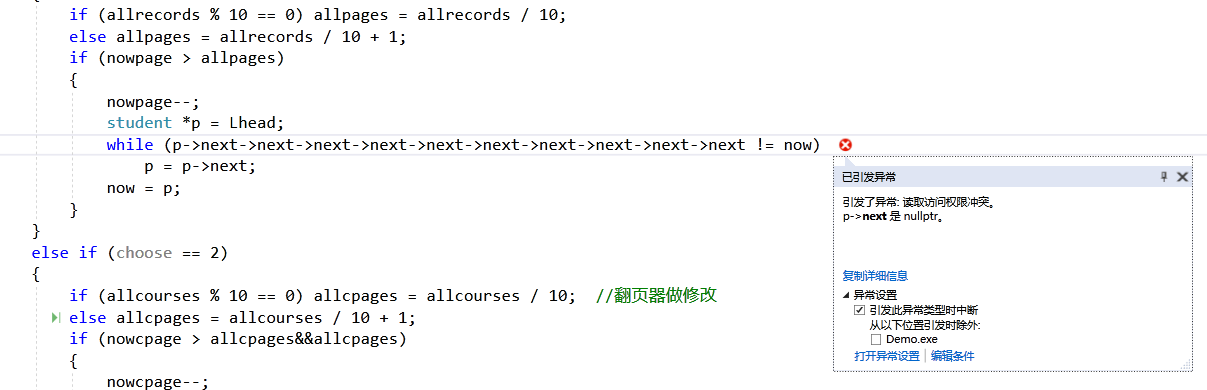
* 1. 编译安装运行说明

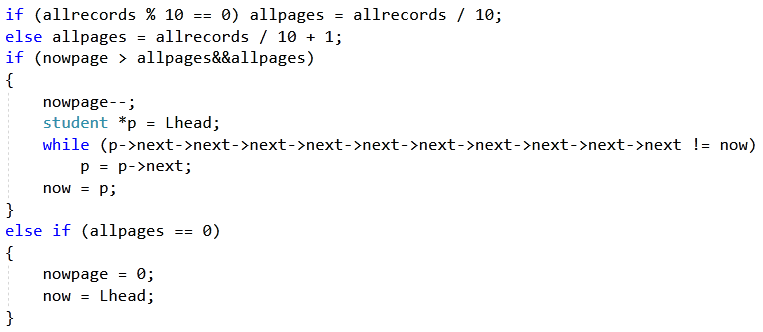
在DEV选择新建项目，在项目下添加三个文件夹，第一个文件夹包含libgraphics所有的.c和.h文件，第二个文件夹包含simpleGUI所有的.c和.h文件，第三个文件夹包含其他自定义.c和.h文件，之后编译运行，系统自动生成.exe文件。

运行.exe文件后，请首先点击左上角的File菜单下的New按钮或是右上方的New按钮新建列表，或者打开已有的文件（范例文件student）。可以增加、删除、修改学生信息，或是搜索特定学生、对已有学生进行排序。不过要注意，在输入学生的信息前请务必保证所填的课程存在。同理，也可以对课程进行增加、删除、修改、搜索操作。退出前请点击左上角的File菜单下的Save按钮或是右上方的Save按钮保存以便下次打开。状态栏统一显示在界面左下角，界面右下角显示文件打开栏与部分选项栏。

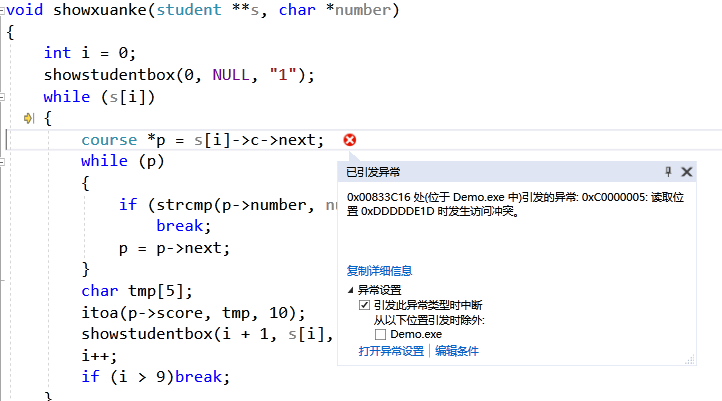
* 1. 典型测试情况

（1）

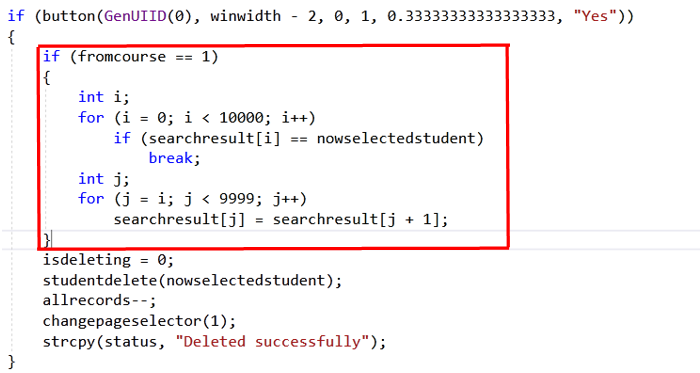


当学生记录只有一条时，删掉这一条就会报错，原因是student \*p是空指针；另外，如果本来有50条记录，我们删掉一条，翻页器就会显示5/4。很显然这与我们的设计初衷是不符合的。经过仔细的分析之后，我们修改了这段代码：

这段代码通过if判断nowpage和allpages的大小和allpages是否为零，防止student \*p为空指针却指向next情况的发生，同时也避免了删除后总页数显示不正确，很好地解决了该问题。

（2）

当我们在选课名单中删除一名学生后，出现了上面这种错误。不难看出，由于删除了该学生结点，searchresult[]中有一个结点不存在了，所以引发了这种错误，所以我们要在确认删除的时候将该学生结点从searchresult[]中删去，并将其后的所有结点往前移一位，所以我们加入了以下一段代码：



这一段代码很好地解决了以上问题。

# 5.组内分工

5.1 组内分工情况

李家豪：主要负责代码编写和调试

朱佳鹏：问题的分析和报告撰写

朱亿江：主要负责资料查阅及报告撰写

5.2个人实践过程中遇到的难点及解决方案

朱佳鹏：

（1）课程本身的难点比较多，遇到问题想要解决的话上网查询往往很难精确找到自己想要的结果，课程对自学的要求很高

（2）对于问题的拆解存在困难，一个复杂的问题分解成多个方面再一一解决，在这点上还做的不好

李家豪：

（1）写代码时比较粗心，有些细节地方出错（==写成=），花了很多时间去调试解决；

（2）有些库函数原型有时候会忘记，需要上网查找；

（3）多文件编译时一开始总是出错，后面慢慢厘清各文件间的关系，逐层包含，解决了这一问题；

（4）链表操作容易造成空指针指向未知区域，需要不断地观察和调试。

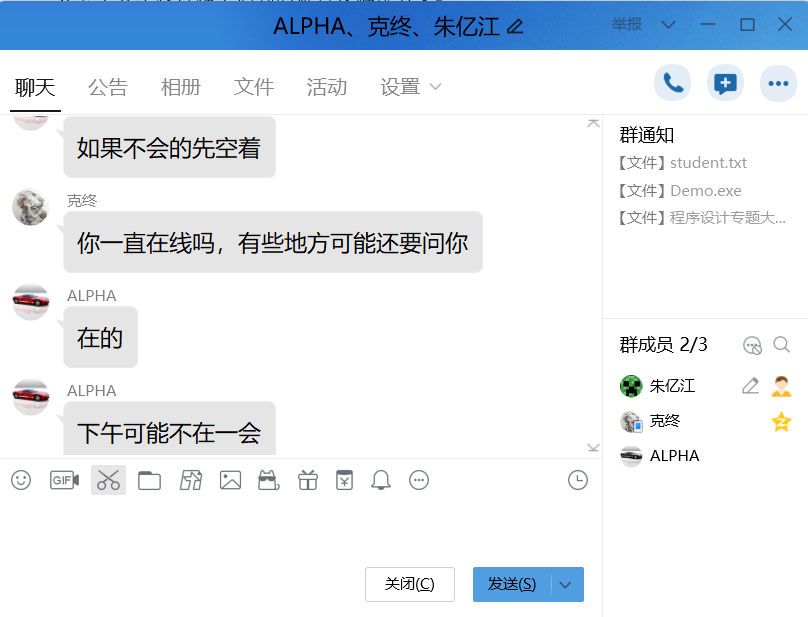
朱亿江：

（1）经常无法理解其余同学编写代码中的要点，需要询问同学或是上网查阅

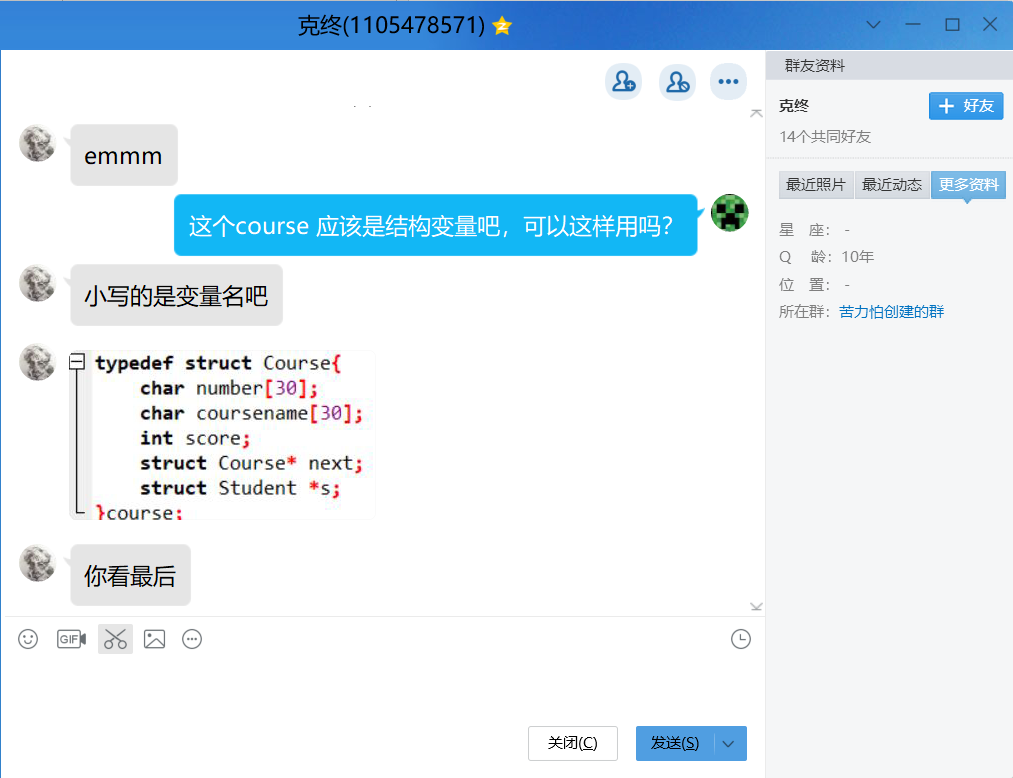
（2）在查阅资料的时候发现资料本身经常存在不断互相引用的情况，一些知识点无法得到很好的理解

# 6.合作纪要

2019年6 月 3日：



2019年6 月4 日：



# 7.总结

7.1程序亮点或创新之处

添加了学生及课程搜索工具，能够更快帮助使用者找到目标。

界面设计简洁，交互功能好，容易上手使用。

7.2应用知识点

字符串比较函数，字符串和数字互相转换的函数，文件的读写和文件格式的定义，链表排序的实现，多文件编译，图形库的应用…

7.3 心得体会

**总体描述**

这次的大作业对小组里的每一个人来说都是一次宝贵的经历，大家都意识到了自身的不足与团队合作的重要性，每个人都得到了进步，更重要的是对编程产生了更加浓厚的兴趣与更加深刻的理解。

朱亿江：这次作业我觉得对我帮助很大。首先是让我看到了自身的不足，有很多别的同学习以为常的知识点是需要我上网查阅资料后才知晓的，而这些知识点往往是被广泛应用的。其次是让我看到了团队合作的重要性，在理解代码和编写报告的过程中，我时常会遇到各种问题，而小组内的其他成员都给了我很多的帮助，没有他们的帮助我是绝对无法完成这份作业的。最后是让我看到了c语言巨大的可能性，一些习以为常的普通函数排列组合形成了许许多多我想都没想过的用法，给了我巨大的启发。

李家豪：这次大作业让我初步了解了软件的构成和开发过程，从一开始的构思到中间的编写，再到最后的调试和运行，每一个环节都是十分重要的。在这过程中我们小组也碰到了各种各样的问题，但经过细心调试之后，所有的问题都被圆满地解决了，这也让我深刻地意识到了软件的开发是一个需要团队合作的任务，只有合理分工，互相配合，才能开发出一款好的软件。除此之外，这次大作业的编写也让我的写代码能力提升了不少，激发了我继续学习这方面知识的兴趣。

朱佳鹏：这次作业从信息录入系统、俄罗斯方块、简易CAD中三选一，每一个任务都不是很简单。在拿到问题之初最令我们头疼的是如何将这个问题化解为若干模块进行解决。其次在报告撰写中也遇到了不少问题。可能因为在解题方式交流上的欠缺导致在解读程序时对程序的许多地方有所不理解。同时也感受到如果没有在编程初期打好框架，在编程过程中乃至后期会出现各种各样的bug等等，而且难以解决。此外，从这次作业中我也理解了为什么C语言能在编程语言当中如此具有统治力的原因，那就是它的各个要素十分基础，从而能外加各种辅助程序来实现某种功能，这使得编写过程和代码本身变得灵活多变。

# 8、参考文献和资料

C语言程序设计（第3版）

<http://c.biancheng.net/>

<https://www.icourse163.org/learn/ZJU-200001?tid=1206289220#/learn/content>

中国大学MOOC——C语言程序设计进阶（翁恺）