**School of Computer Science and Technology, Huazhong University of Science and Technology**

**C language course design report**

topic: Scientific research project information management system

Major: Computer Science and Technology

class class: Accounting ( hand in ) 1601

study No.: U201612696

surname Name: Chen Haorui

become Achievement :

Instructor : Gan Zaobin

**Completion date: August 23 ,** 2017

Table of contents

[1. System requirements analysis](#_Toc493364555)  [3](#_Toc493364555)

[1.1 Basic information entry, modification and deletion functions](#_Toc493364556)  [3](#_Toc493364556)

[1.2 Basic Information Query Function](#_Toc493364557)  [3](#_Toc493364557)

[1.3 Data statistics function](#_Toc493364558)  [2](#_Toc493364558)

[1.4 Data access function](#_Toc493364559)  [2](#_Toc493364559)

[1.5 Display system-related information function](#_Toc493364560)  [2](#_Toc493364560)

[2. Overall Design](#_Toc493364561)  [3](#_Toc493364561)

[2.1 File operation module](#_Toc493364562)  [3](#_Toc493364562)

[2.2 Data maintenance module](#_Toc493364563)  [4](#_Toc493364563)

[2.3 Data query module](#_Toc493364564)  [4](#_Toc493364564)

[2.4 Data statistics module](#_Toc493364565)  [5](#_Toc493364565)

[2.4 Data statistics module](#_Toc493364566)  [5](#_Toc493364566)

[3. Data structure design](#_Toc493364567)  [6](#_Toc493364567)

[3.1 System data](#_Toc493364568)  [6](#_Toc493364568)

[3.1.1 Basic Information of Schools and Departments](#_Toc493364569)  [6](#_Toc493364569)

[3.1.2 Basic information of scientific research team](#_Toc493364570)  [6](#_Toc493364570)

[3.1.3 Basic information of scientific research projects](#_Toc493364571)  [7](#_Toc493364571)

[3.2 Data Organization](#_Toc493364572)  [7](#_Toc493364572)

[3.2.1 Department node composition](#_Toc493364573)  [8](#_Toc493364573)

[3.2.2 Research Team Node Composition](#_Toc493364574)  [8](#_Toc493364574)

[3.2.3 Node composition of scientific research projects](#_Toc493364575)  [9](#_Toc493364575)

[3.2.4 Project Leader Node Composition](#_Toc493364576)  [10](#_Toc493364576)

[3.3 Data structures](#_Toc493364577)  [10](#_Toc493364577)

[Fourth, detailed design](#_Toc493364578)  [12](#_Toc493364578)

[4.1 File Management Section](#_Toc493364579)  [12](#_Toc493364579)

[4.1.1 File entry](#_Toc493364580)  [12](#_Toc493364580)

[4.1.2 File output](#_Toc493364581)  [13](#_Toc493364581)

[4.1.3 Data Backup](#_Toc493364582)  [14](#_Toc493364582)

[4.1.4 Data Recovery](#_Toc493364583)  [15](#_Toc493364583)

[4.2 Data Maintenance Section](#_Toc493364584)  [16](#_Toc493364584)

[4.2.1 Department data entry](#_Toc493364585)  [16](#_Toc493364585)

[4.2.2 Data entry of scientific research team](#_Toc493364586)  [17](#_Toc493364586)

[4.2.3 Scientific research project data entry](#_Toc493364587)  [17](#_Toc493364587)

[4.2.4 Department data revision](#_Toc493364588)  [18](#_Toc493364588)

[4.2.5 Research team data modification](#_Toc493364589)  [19](#_Toc493364589)

[4.2.6 Data modification of scientific research projects](#_Toc493364590)  [21](#_Toc493364590)

[4.2.7 Department data deletion](#_Toc493364591)  [22](#_Toc493364591)

[4.2.8 Deletion of research team data](#_Toc493364592)  [22](#_Toc493364592)

[4.2.9 Data deletion of scientific research projects](#_Toc493364593)  [23](#_Toc493364593)

[4.3 Information Finding Section](#_Toc493364594)  [23](#_Toc493364594)

[4.3.1 Searching for Faculty and Department Information](#_Toc493364595)  [23](#_Toc493364595)

[4.3.2 Research team information search](#_Toc493364596)  [24](#_Toc493364596)

[4.3.3 Research project information search](#_Toc493364597)  [25](#_Toc493364597)

[4.4 Information statistics part](#_Toc493364598)  [25](#_Toc493364598)

[4.4.1 Teacher-student ratio](#_Toc493364599)  [26](#_Toc493364599)

[4.4.2 Number of items](#_Toc493364600)  [26](#_Toc493364600)

[4.4.3 Number of National Natural Science Foundation of China projects](#_Toc493364601)  [27](#_Toc493364601)

[4.4.4 The ratio of the number of team projects to the number of teachers](#_Toc493364602)  [28](#_Toc493364602)

[4.4.5 Number of projects undertaken by the team leader](#_Toc493364603)  [30](#_Toc493364603)

[4.5 Help Section](#_Toc493364604)  [30](#_Toc493364604)

[4.5.1 Help topics](#_Toc493364605)  [30](#_Toc493364605)

[4.5.2 System Copyright Description Topic](#_Toc493364606)  [31](#_Toc493364606)

[V. System Realization](#_Toc493364607)  [32](#_Toc493364607)

[5.1 dorm.h](#_Toc493364608)  [32](#_Toc493364608)

[5.2 main.c](#_Toc493364609)  [39](#_Toc493364609)

[6. Running test and result analysis](#_Toc493364610)  [151](#_Toc493364610)

[6.1 Interface and window operation](#_Toc493364611)  [151](#_Toc493364611)

[6.2 Operation status of data maintenance function](#_Toc493364612)  [153](#_Toc493364612)

[6.3 Operation status of data query function](#_Toc493364613)  [164](#_Toc493364613)

[6.4 Operation of data statistics function](#_Toc493364614)  [168](#_Toc493364614)

[6. System information function running status](#_Toc493364615)  [171](#_Toc493364615)

[7. Summary](#_Toc493364616)  [172](#_Toc493364616)

[8. References](#_Toc493364617)  [174](#_Toc493364617)

# 1. System Requirements Analysis

The scientific research information management system is used to manage the scientific research related information of the departments, mainly including the basic information of the departments, the basic information of the scientific research team, and the basic information of the scientific research projects, so as to help the relevant leaders to keep abreast of the scientific research tasks of each department and team, so as to Promote the coordinated development of various disciplines.

The scientific research data management system mainly realizes the following functions :

## 1.1 Basic information entry , modification and deletion functions

The basic new information of the scientific research project management system mainly includes the following three categories :

1. Department information: the name of the department , the person in charge of the department , contact number and other data items.
2. Scientific research team information : team name , team leader, number of teachers, number of graduate students, departments and other data items .
3. Scientific research project information : project number , project category, start time, project leader , team and other data items.

, modification and deletion functions of the above three basic data information . When entering information , the system should provide as fast and convenient a data entry method as possible to avoid repeated operations and reduce data redundancy.

## 1.2 Query function of basic information

The system should realize the query function of the above three basic data information, and provide methods for querying according to various conditions , including:

1. Query the department information by the person in charge of the department ;
2. According to all or part of the name of the department to query the information of the department;
3. Query the basic information of the scientific research team according to all or part of the team name;
4. Query the basic information of the scientific research team according to the upper and lower limits of the number of teachers;
5. by project number;
6. Query the basic information of scientific research projects according to the team they belong to;

In the above query results, if multiple pieces of information are found, a function of displaying them one by one should be provided.

## 1.3 Data statistics function

On the basis of the above three basic data information , various data statistics functions are provided, including:

(1) Count the total number of teachers in each department, the total number of graduate students, and the ratio of graduate students to teachers (reserve 2 decimal places), according to the number of students and teachers

After the number ratio is sorted in descending order, the statistical results are output.

(2) Count the number of scientific research projects of each department in a certain year, the number of 973 projects, the number of 863 projects, and the total funding of scientific research, and the number of scientific research projects in descending order

output in sequence.

(3) Statistics of the top 10 scientific research teams with the largest number of scientific research projects in the category of National Natural Science Foundation of China over the years, sorted in descending order of the number of projects

Then output the name of the scientific research team, the number of scientific research projects of the National Natural Science Foundation of China, and the total amount of project funds.

The 5 scientific research teams with the highest ratio of the number of scientific research projects to the number of teachers are counted , sorted in descending order according to the ratio (with two decimal places),

Output the name of the scientific research team, the number of teachers, the number of scientific research projects, and the ratio between the number of projects and the number of teachers.

(5) Count the number of projects undertaken by each project leader in the selected department , sort in descending order of the number, and output the name of the person in charge and the number of projects undertaken.

## 1.4 Data access function

The above three kinds of information exist in the memory in the form of a linked list structure when the program is running, and the data is stored using a dynamic storage allocation method. At the same time , the data is stored in the form of data files on the hard disk, and the consistency of the content of the data on the two storage media of the memory and the hard disk is guaranteed. Its functions specifically include :

imported every time the system is turned on , and the function of printing the report (to file) is provided in the main menu , and the latest data are stored in the dat file and the txt file respectively . The dat file is used for the next data import, and the txt file is convenient for data review ; At the same time, it provides data backup and data recovery functions , and the files exported by the data backup function are the current system time.

## 1.5 Display system-related information function

The system should listen to relevant information about itself to the user to prompt the operating environment and perform subsequent updates of the software . At the same time, there should be instructions about the use of the system . Specifically include :

1. Open the function of the system use guidance file;
2. The function of displaying system-related information ;

# 2. Overall design

The scientific research project management system consists of five modules : file module , data maintenance module , data query module , data statistics module and help module , as shown in Figure 2-1 .

Figure 2-1 Functional modules of scientific research project information management system

The functions of these five modules and their sub-modules are described below .

## 2.1 File operation module

The functions of the file module include a series of environmental maintenance and data assurance operations related to system start-up and system end-run . It is further divided into 6 sub-modules: interface initialization , data loading , data saving, data backup , data recovery , and exiting the system as shown in Figure 2-2 .

Figure 2-2 File Module Subsystem

1. Interface initialization sub-module : used to set the display mode of the console window , set the screen window as a text interface with 50 columns and 100 lines, set the window title bar, clear the screen and display the system menu bar and system status bar .
2. Data loading sub-module : used to read the basic data stored in multiple data files into memory, construct a data link list, and output data loading related prompt information at the same time .
3. Data saving sub-module : used to save the linked list data in memory to each data file respectively according to the default path.
4. Data backup sub-module : used to dump the basic data stored in multiple data files into a data file according to the path specified by the user , for the user to restore the system after the system data is damaged or lost.
5. Data recovery sub-module : corresponding to the function of the data backup sub-module, it can recover the system data at the backup time point from a backup data file, and load the recovered data into the memory. Together with the data backup sub-module , the data recovery sub-module is used to improve the security and reliability of the system.
6. Exit the system sub-module : release the dynamic storage area during program running, close the console standard input and output device handles , set the title bar to " running finished " , clear the screen window information, and end the system running.

## 2.2 Data maintenance module

the entry , modification and deletion of system code information and basic data information to ensure the accuracy, integrity and validity of the data. This module is divided into : departmental information maintenance , scientific research team information maintenance, scientific research project information maintenance . The department information maintenance is further divided into : input, modification and deletion of department information; input, modification and deletion of scientific research team information; input, modification and deletion of scientific research projects , as shown in Figure 2-3 .

These sub- modules are respectively used to enter, modify and delete departmental information , basic information of scientific research teams and basic information of scientific research projects, so as to ensure the accuracy of these three basic data . At the same time , these three kinds of basic data are also stored in the data link list , and are respectively saved in the department information data file, scientific research team information data file and scientific research project information data file , so as to keep the data content on the two storage media of memory and hard disk. consistency.

Figure 2-3 Data maintenance module subsystem

## 2.3 Data query module

query functions for system code information and three kinds of data information . This function consists of three sub-modules : department information query module , scientific research team information query module , and scientific research project information query module, as shown in Figure 2-4 .

Figure 2-4 Data query module subsystem

1. Department information query sub-module : used to provide the function of querying department information according to the person in charge of the department or all or part of the name of the department;
2. the function of querying the basic information of the scientific research team by all or part of the team name or by the upper and lower limit of the number of teachers ;
3. It is used to provide the function of querying the basic information of scientific research projects by project number or by the team they belong to .

## 2.4 Data statistics module

The data statistics module provides the function of multi-faceted statistics on the three basic data . According to statistical conditions , this module is divided into teacher-student ratio, total number of projects , number of science fund projects, project /teacher ratio , and project allocation, as shown in Figure 2-5 .

Figure 2-5 Data statistics module subsystem

1. Teacher-student ratio sub-module: It is used to count the total number of teachers in each department, the total number of graduate students, and the ratio of graduate students to teachers (retain 2 decimal places), and output the statistical results after sorting in descending order according to the ratio of the number of students to teachers.
2. Sub-module of the total number of projects : used to count the number of scientific research projects of each department in a certain year, the number of 973 projects, the number of 863 projects, and the total scientific research funds, and output the number of scientific research projects in descending order.
3. Science Foundation project sub-module : used to count the 10 scientific research teams with the largest number of scientific research projects in the category of National Natural Science Foundation of China over the years, and output the name of the scientific research team after sorting in descending order of the number of projects, and the number of scientific research projects in the category of National Natural Science Foundation of China. and the total project cost.
4. Project/teacher ratio sub-module: used to count the 5 scientific research teams with the highest ratio between the number of scientific research projects and the number of teachers , sort them in descending order according to the ratio (with two decimal places), and output the name of the scientific research team, number of teachers, number of scientific research projects, projects ratio to the number of teachers.
5. Project Allocation Sub-module : It is used to count the number of projects undertaken by each project leader in the selected department , and output the name of the person in charge and the number of projects undertaken after sorting the quantity in descending order.

## 2.4 Data statistics module

The helper module provides help information for users to use the system , as well as system version information and system copyright information. It is divided into help topic sub-modules and system copyright description sub-modules , as shown in Figure 2-6 .

1. Help topic sub-module : used to provide users with help information by keyword
2. System version copyright description sub-module: used to display system version information and system copyright information .

# 3. Data structure design

## 3.1 System data

In computers, objective objects and their relationships are described by data. At the same time of system function design , it is necessary to analyze the entity objects and their relationships in the system , establish a data model that describes the entity, and then realize the internal storage structure of data based on this, and then establish a data support environment for system operation.

Entities that the system needs to handle include departments , teams, and projects. Among them, teams and departments rely on the names of departments, and projects and teams rely on team names to establish affiliation relationships . Figure 3-1 describes their ER relationship diagrams .

属于

1

n

1

n

属于

院系

团队

项目

Figure 3-1 Department - Team-Project ER Diagram

According to the task requirements , there are three types of basic information to be processed by the system: department information , scientific research team information and scientific research project information . There is such a relationship between the three types of information : the department name in the department information is related to the department name in the scientific research team information, and the team name in the scientific research team information is related to the team name in the scientific research project . These three kinds of basic information become the basic data of the system through input .

### 3.1.1 Basic information of departments

The department information table provides the department entity in the system, which mainly includes 3 parts. These 3 parts are the name of the department, the name of the person in charge, and the contact number. The types of the above 3 parts in the system are char type Array, where the length of the city number array is 20, the length of the city name array is 12, and the length of the city level array is 15. The following table is the basic information table of the city:

|  |  |  |
| --- | --- | --- |
| Chinese field name | type and length | example or description |
| Department name | char school\_name [20] | Society |
| principal | char leader[12] | Ding Jianding |
| contact number | char contact\_num [15] | 87543152 |

Table 3 - 1 Basic Information of Schools and Departments

### 3.1.2 Basic information of the scientific research team

The scientific research team information table provides the entity of the scientific research team in the system, and it mainly includes 5 parts. These 5 parts are the team name, the name of the person in charge, the number of teachers, the number of graduate students, and the name of the department. Among them, the team name, the name of the person in charge, and the name of the department are char arrays in the system , and the array length of the team name is 30. The array length of the name of the person in charge is 12, and the array length of the name of the department is 20; The types of the number of people and the number of graduate students in the system are all int types. The following table is the basic information table of the scientific research team:

|  |  |  |
| --- | --- | --- |
| Chinese field name | type and length | example or description |
| Team Name | char team\_name [30] | soc7 |
| name of person in charge | char leader[12] | ewr |
| Number of teachers | int teacher\_num | 4 |
| Number of graduate students | int stu\_num | 7 |
| Faculty | char school\_name [20] | Society |

Table 3-2 Basic Information of Scientific Research Team

### 3.1.3 Basic information of scientific research projects

The scientific research project information table provides the scientific research project entity in the system, which mainly includes 6 parts. These 6 parts are project number, project category, start time, project funding, person in charge name, team name, among which the project number , start time, project funds, and team name in the system are all char arrays, and the array length of the project number is 15. The array length of the start time is 8, the array length of the name of the person in charge is 12, and the team name The length of the array is 3 0; the type of project category in the system is char type, and the type of project funds in the system is float type. The following table is the basic information table of scientific research projects:

|  |  |  |
| --- | --- | --- |
| Chinese field name | type and length | example or description |
| Item Number | char project\_num [15] | 744482 |
| Item category | char type | 2 |
| start time | char open\_time [8] | 2008\12 |
| project funding | char jingdianmingcheng [15] | 4 2.534 |
| name of person in charge | char leader[12] | wub |
| Team name | char team\_name [30] | soc7 |

Table 3-3 Basic information table of scientific research projects

## 3.2 Data Organization

in 3.1 , the system establishes three logical tables of basic information of departments, basic information of scientific research teams, and basic information of scientific research projects . These three tables represent the data views presented to users by the system . In the system implementation , the data structure of memory linked list and disk file headphone is used to store these logical views.

The user enters relevant information through the form interface , and the system converts the form information into a memory linked list, and then converts the linked list into a disk file form. For the user's service request , the system first transfers the disk file data into the memory linked list, and then responds to the user's request by processing the linked list's data.

The system uses three binary files of basic information files of departments, basic information files of scientific research teams, basic information files of scientific research projects, and a full information text file as disk storage corresponding to the form, and stores three types in the memory in the form of structural variables. Basic information, and organize all data in the form of a linked list.

### 3.2.1 School and Department Node Composition

Department information is described in the structure using the structure type . The members of the departmental structure include : the name of the department, the person in charge of the department, the contact information, the total number of projects under the department, the number of 973 projects, the number of 863 projects, the total number of teams, the total number of teachers in each team, the total number of graduate students, the ratio of the number of graduate students to teachers ,total budget. There are two structure pointers in the department structure , pointing to the next department structure and scientific research team structure respectively. Through these two pointers, all the departments of the school and all the scientific research teams of a department can be linked to form a linked list of departments ( Level 1 ) and the linked list of scientific research teams under the departments ( level 2 ) . This structure variable realizes the storage of the department information table in the memory structure , and its structure declaration is shown in Figure 3-2 .

Data structure in memory : stored on the main chain node of the cross-linked list, each node not only saves the address of the next node, but also saves the head node address of the basic information list of the scientific research team under the department , as shown in the figure 3-6 shown . \_

typedef struct school\_info{

char school\_name[20];

char leader[12];

char contact\_num[15];

unsigned short num\_proj;

unsigned short num\_973;

unsigned short num\_863;

float sum\_teacher;

float sum\_student;

float proportion;

float budget;

struct school\_info \*next;

struct team\_info \*team\_head;

}school;

Storage structure of data files : each piece of information is stored in a binary file as a record.

Figure 3-2 Membership table of faculty structure

### 3.2.2 Node composition of scientific research team

The research team uses the structure type to describe in the structure . The members of the department structure include : team name, team leader, department name, number of teachers, number of graduate students, number of scientific research projects, number of projects of the National Natural Science Foundation of China, total funding of the projects of the National Natural Science Foundation of China, ratio of scientific research projects to the number of teachers, number of research projects. There are two structure pointers in the scientific research team structure , which point to the next scientific research team structure and scientific research project structure respectively. Through these two pointers, all the teams, all departments under a certain department and all the scientific research projects of a team can be connected respectively to form a Linked list of scientific research team ( level 2 ) and linked list of scientific research projects under the scientific research team ( level 3 ) . This structure variable realizes the storage of the scientific research team information table in the memory structure , and its structure declaration is shown in Figure 3-3 .

Data structure in memory : stored in the scientific research team information node of the cross-linked list, each node not only saves the address of the next node, but also saves the head node address of the basic information list of scientific research projects under the department , as shown in the figure 3-6 shown . \_

Storage structure of data files : each piece of information is stored in a binary file as a record.

Figure 3-3 Team structure member table

typedef struct team\_info{

char team\_name[30];

char leader[12];

char school\_name[20];

int teacher\_num;

int stu\_num;

unsigned short proj\_num;

unsigned short projsci\_num;

float proj\_budget;

float proportion;

struct team\_info \*next;

struct project\_info \*proj\_head;

}team;

### 3.2.3 Node composition of scientific research projects

The scientific research project is described in the structure using the structure type . The members of the scientific research project structure include : project number, project type, start time, project leader, team name, and project budget. In the scientific research project structure, there is a structure pointer pointing to the next scientific research team structure . Through this pointer, all scientific research projects under a certain team can be connected to form a linked list of scientific research projects under the scientific research team ( three levels ) . This structure variable realizes the storage of the scientific research project information table in the memory structure , and its structure declaration is shown in Figure 3-4 .

Data structure in memory : stored in the scientific research team information node of the cross-linked list, each node saves the head node address of the next scientific research project basic information linked list , as shown in Figure 3-6 .

Storage structure of data files : each piece of information is stored in a binary file as a record.

Figure 3-4 project structure member table

typedef struct project\_info{

char project\_num[15]；

char type；

char open\_time[8]；

char leader[12]；

char team\_name[30];

float budget;

struct project\_info \*next;

### 3.2.4 Project Leader Node Composition

The information of the person in charge of the scientific research project is described in the structure using the structure type . The structural members of the person in charge of scientific research projects include : the name of the team leader, and the number of projects in charge. In the scientific research project leader structure, there is a structure pointer pointing to the structure of the next scientific research team leader . Through this pointer, all the scientific research project leaders under a certain department can be linked to form a linked list of scientific research project leaders ( independent of the three-level cross outside of the cross-linked list ) . This structure variable realizes the storage of the information table of the person in charge of the scientific research project in the memory structure , and its structure declaration is shown in Figure 3-5 .

ttypedef struct leader{

char leader[12];

unsigned short n;

struct leader \*next;

}lead;

Data structure in memory : Stored in the nodes of the person-in-charge information one-way linked list .

Storage structure of data files : not stored in files.

Figure 3-5 project leader structure member table

## 3.3 Data structure

The linked list used in this system is a cross-linked list in three directions plus an independent linked list that is only established in a specific function . The first layer chain is the basic information chain of the department, the second layer chain is the basic information chain of the scientific research team, and the third chain is the basic information chain of the scientific research project. This linked list is automatically created according to the information imported from the disk when entering the system, and the life cycle is synchronized. and the operation of the system ; the single-layer independent linked list only stores the basic information of the team leader , and its life cycle is the function that creates it.

As long as the system enters the number corresponding to the corresponding function on the keyboard and presses the Enter key, the system will automatically jump to the corresponding function interface, and the user can perform corresponding operations.

Figure 3-6 Three -way cross-linked list

**…**

head

**…**

团队11

团队12

团队1q

院系1

院系2

院系p

项目111

**…**

项目11r

∧

∧

∧

# 4. Detailed design

## 4.1 File management part

### 4.1.1 File entry

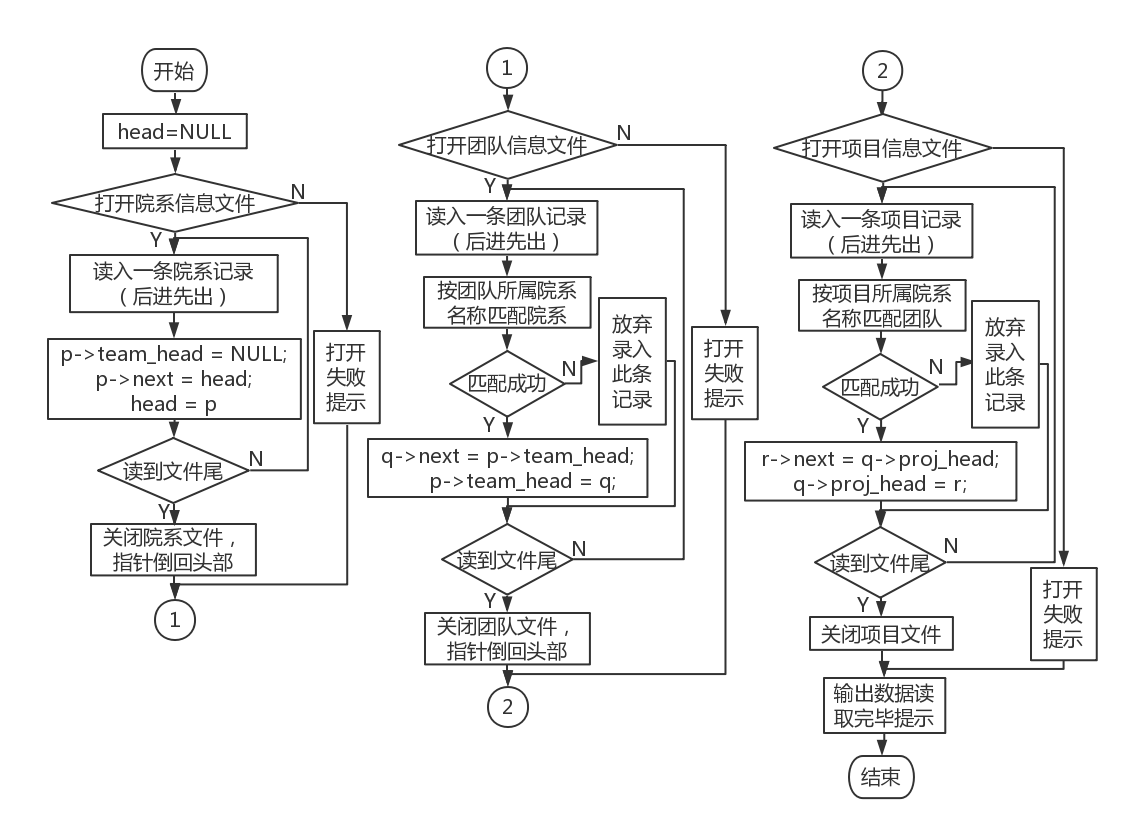


Figure 4-1 File entry flow chart

1. Department pointer is empty
2. Open the department information binary file
3. Read in a department record in last-in-first-out
4. Close the department file, and the pointer will go back to the head
5. Open the team info binary
6. Match departments by department names read into team records
7. Close the team file , the pointer goes back to the head
8. Open the project info binary
9. Match departments by department names read into project records
10. Close the project file , the pointer goes back to the head

### 4.1.2 File output

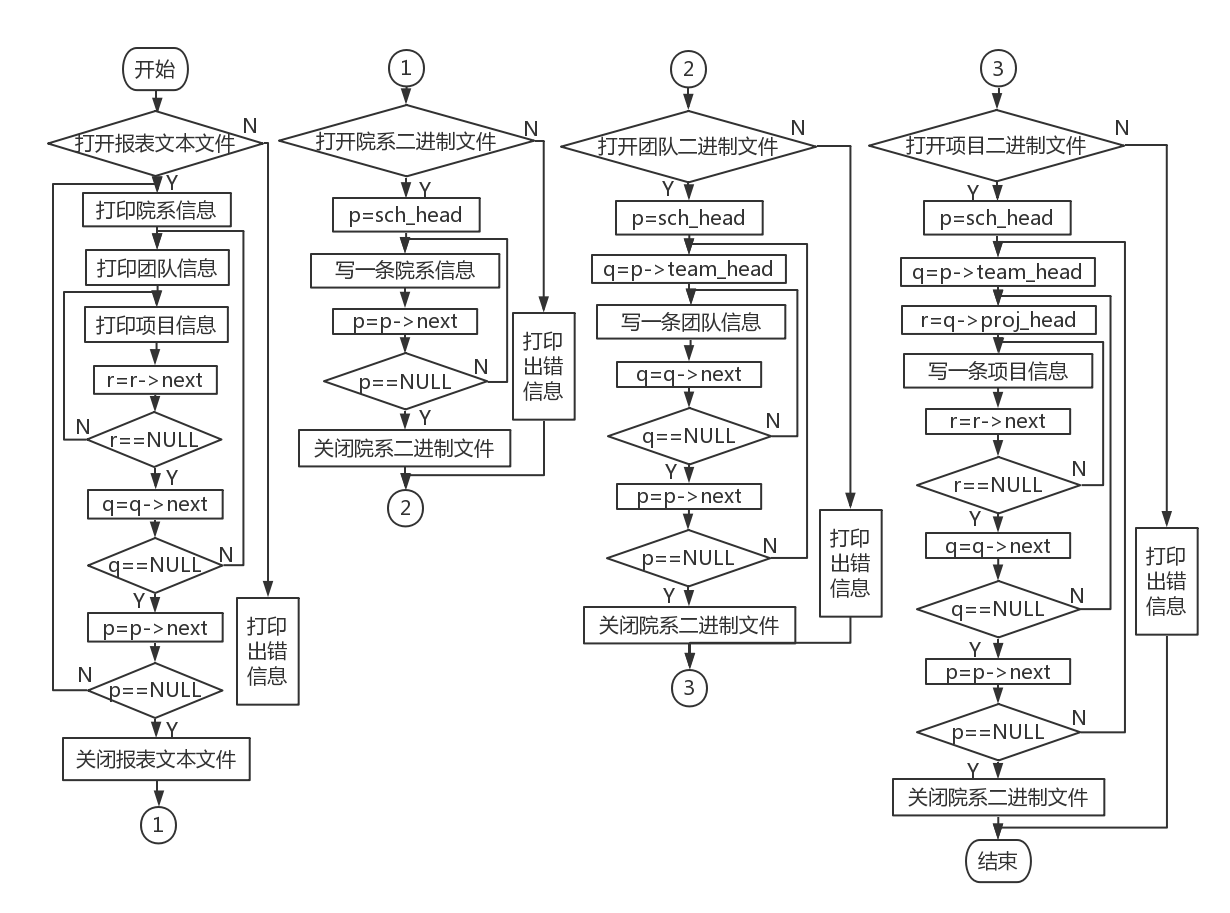


Figure 4-2 \_ File saving flow chart

1. Open report text file
2. Print a department record
3. print a team record
4. Print all project records under the team
5. Return to 3 after printing all project records under the team
6. under the department will be printed and returned to 2
7. close report text file
8. Open the department binary
9. Cyclic write department records
10. Close Faculty Binaries
11. Open the team binary
12. Double loop write team records
13. close team binaries
14. Open the project binary
15. Triple loop write item records

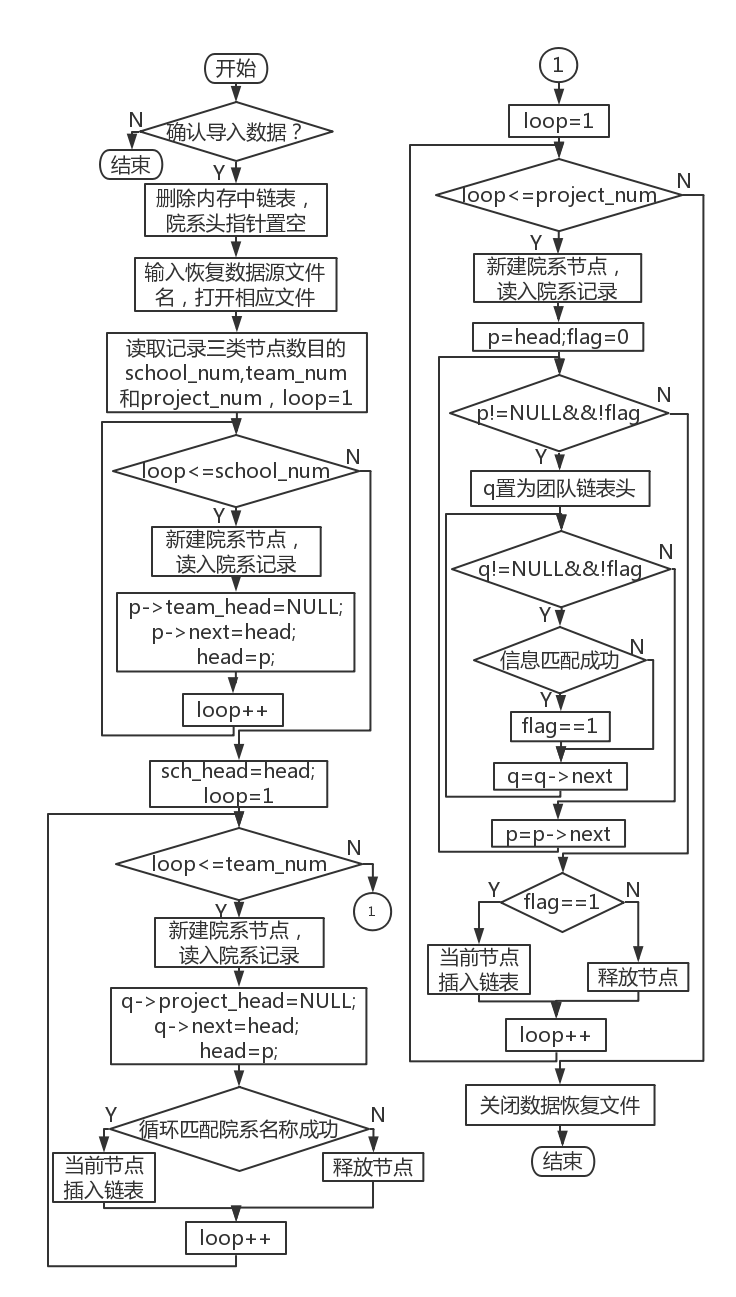
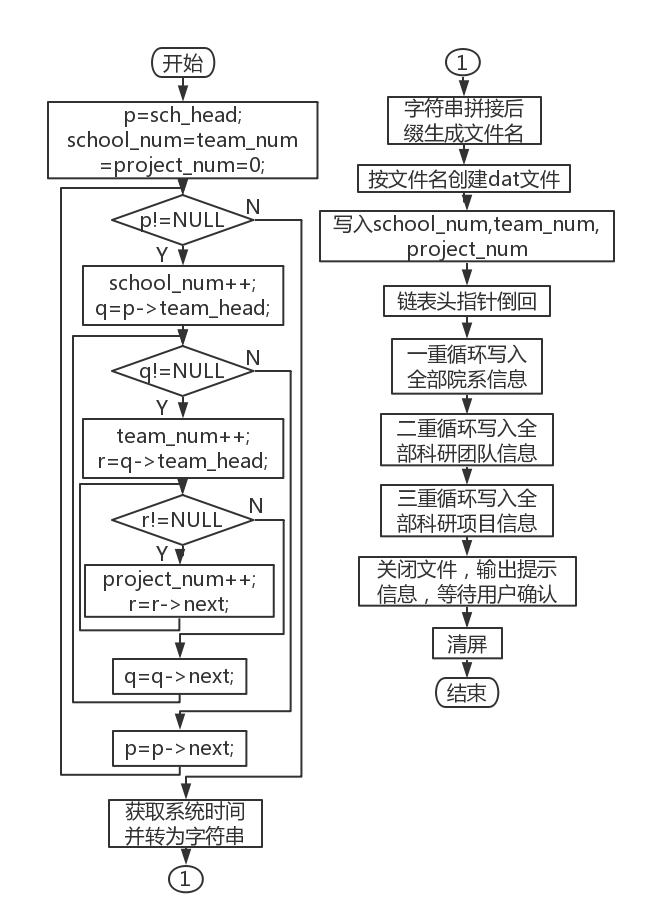
16. Close the project binaries

Figure 4-3 \_ Data backup flowchart Figure 4 - 4 Data Recovery Flowchart

### 4.1. 3 Data backup

1. The three variables for counting the number of nodes are set to 0
2. Triple cycle statistics of the number of various nodes
3. Get the current system time
4. Concatenated with the file type suffix to form the full file name
5. Create binary file by filename and open
6. Write three statistical variables
7. Enter the department structure according to the number of school\_num
8. the scientific research team structure according to the number of team\_num
9. Cyclic input of scientific research project structure according to the number of project\_num
10. Close the file and output a prompt message

### 4.1. 4 Data recovery

1. Perform a second confirmation of the operation

2. Clear the original linked list and set the head pointer to empty

3. Enter the recovery data target file

4. Read in the statistical variables of the three types of nodes

5. Enter the department information node with school\_num as the cycle upper limit

6. The head pointer of the scientific research team is empty, and the head pointer of the department moves to the current node

7. Head pointer rewinds

8. Enter the scientific research team information node with team\_num as the upper limit of the cycle

9. The head pointer of the scientific research team is empty

10. Cyclic comparison of matching information, if successful, insert , otherwise delete the current node

11. Head pointer rewinds

12. Enter the scientific research team information node with project\_num as the upper limit of the cycle

13. The head pointer of the scientific research project is empty

14. Cyclic comparison of matching information, if successful, insert , otherwise delete the current node

15. Close the data recovery file and give a prompt message

## 4.2 Data maintenance part

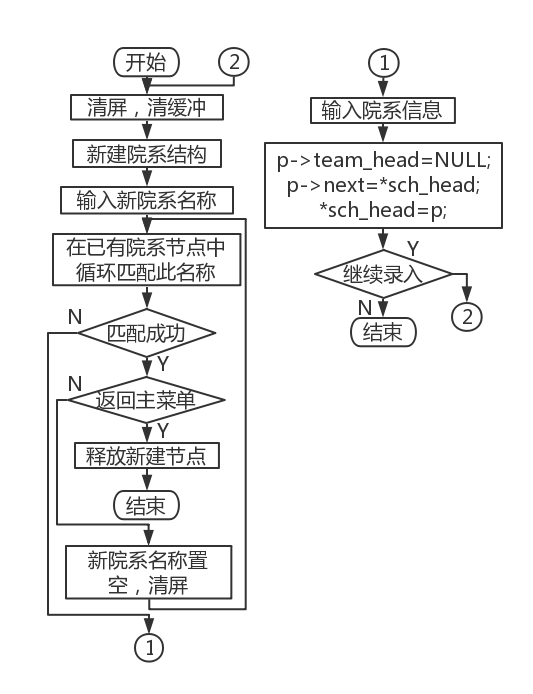
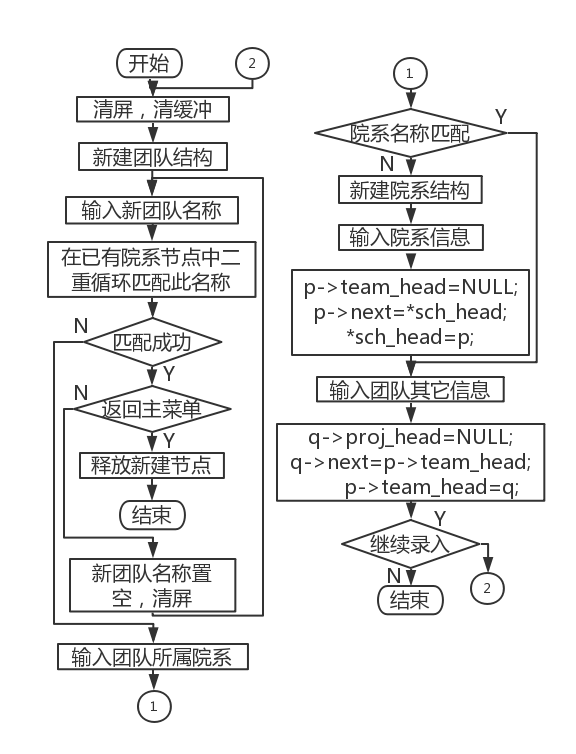
 

Figure 4-5 \_ Faculty entry flow chart Figure 4-6 \_ Team Entry Flowchart

### 4.2.1 Faculty and department data entry

1. The head pointer is empty

2. Clear the console text and clear the buffer

3. Create a new departmental structure

4. Enter the name of the new department

5. Circularly match the names of existing departments , if the matching fails, go to 6, otherwise go to 7

6. Continue if you return to the main menu , otherwise go to 4

7. Release the new node , clear the console , and go to 11

8. Enter other information of the department and the team head pointer is blank

9. The head pointer points to this new node

10. Continue to enter and return to 2, otherwise continue

11. end

### 4.2.2 Data entry of research team

1. New team structure

2. Clear the console text and clear the buffer

3. Enter the new team name

4. The double cycle matches the name of the existing team . If the match fails, go to 7, otherwise go to 5

5. Continue if you return to the main menu , otherwise go to 3

6. Release the new node , clear the console , go to 1 5

7. Enter the department of the team

8. Circularly match the name of the department, if it is successful, go to 12, otherwise continue

9. Create a new departmental structure

10. Enter the department information, and the team head pointer is blank

11. The head pointer points to the newly created node

12. Enter the team information, and the project head pointer is blank

13. The department head pointer points to this new node

14. Continue to enter and return to 2, otherwise continue

15. end

### 4.2. 3 Data entry of scientific research projects

1. New project structure
2. Clear console text and buffer
3. Enter new item number
4. The double cycle matches the name of the existing team , if the match fails, go to 7, otherwise continue
5. If you return to the main menu, continue , otherwise go to 3
6. Release the new node , clear the console , and go to 18
7. Enter the department and team of the project
8. Circularly match the name of the department, if successful, go to 12
9. New departmental structure
10. Enter the department information, the team head pointer is blank
11. The head pointer points to the new node
12. Loop to match the team name , if successful, go to 15
13. New team structure
14. Enter team information, the project head pointer is blank
15. The department head pointer points to this new node
16. Enter project information
17. If you continue to enter , go to 2, otherwise continue
18. Finish

### 4.2. 4 Department data modification

1. Clear screen , clear buffer

2. If it is not the first cycle , enter the error message

3. If you return to the main menu , go to 10, otherwise continue

4. Choose the way to search for departments

5. Enter match target

6. Loop to match the current information, if the match is successful, exit the loop and keep the pointer position , otherwise go to 1

7. Select the modification method to modify the specified information

8. If you continue to modify the information of the department , go to 7, otherwise continue

9. If you continue to modify, go to 1, otherwise continue

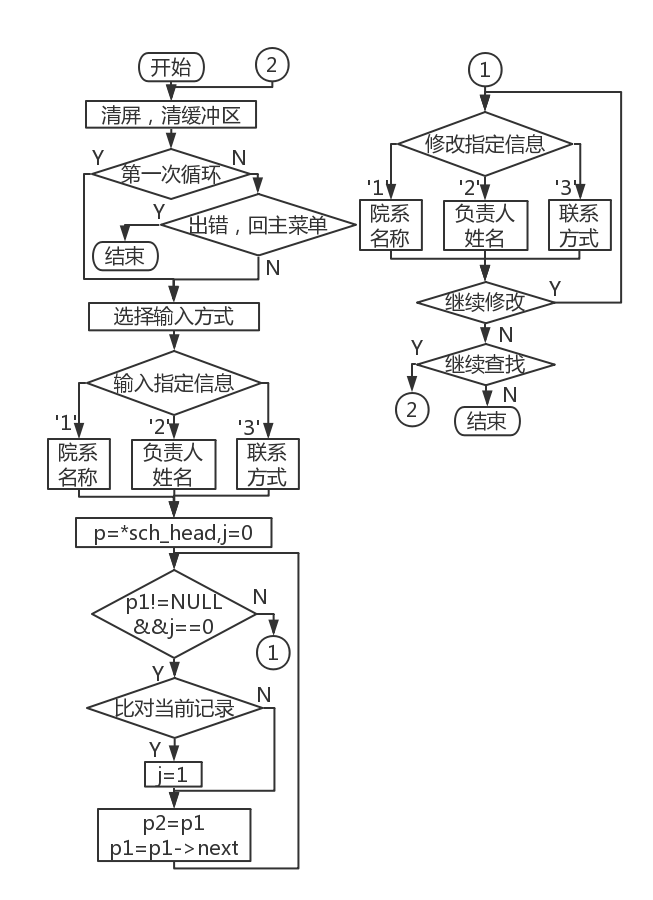
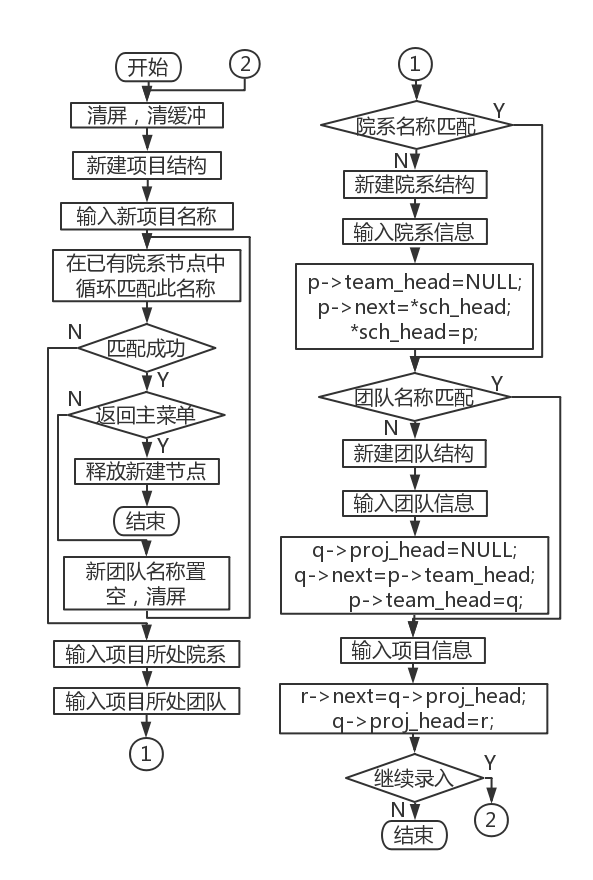
10. end

Figure 4-7 \_ Project Entry Flowchart Figure 4 - 8 Faculty modification flow chart

### 4.2.5 Data modification of scientific research team

1. Clear screen , clear buffer

2. If it is not the first cycle , enter the error message

3. If you return to the main menu , go to 9, otherwise continue

4. Enter the team name

5. The double loop matches the current information. If the match is successful, exit the loop and keep the pointer position , otherwise go to 1

6. Select the modification method to modify the specified information

7. If you continue to modify the department information , go to 6, otherwise continue

8. If you continue to modify, go to 1, otherwise continue

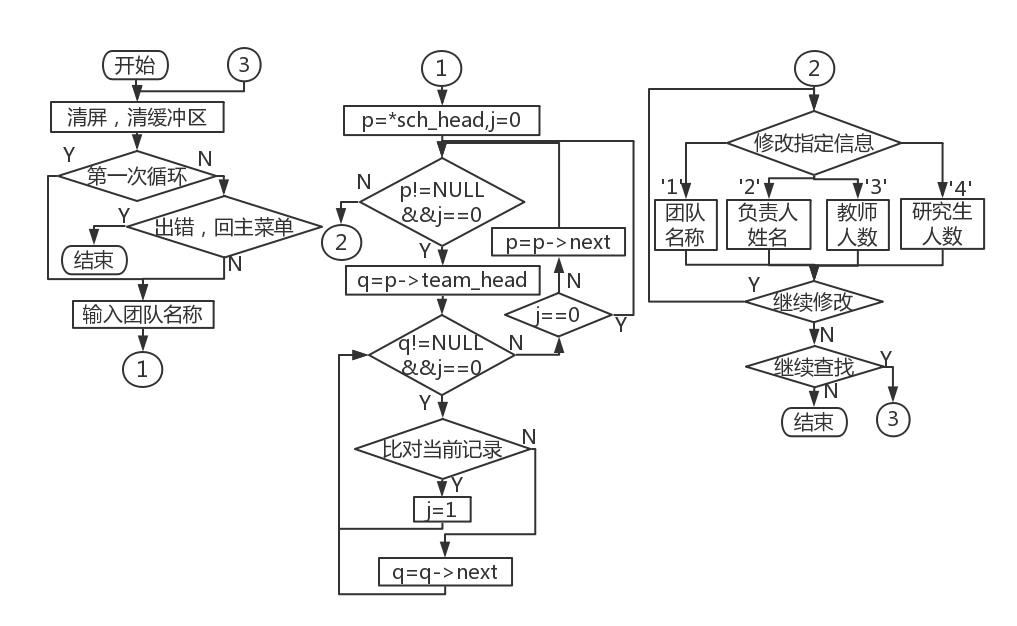
9. end 

Figure 4-9 \_ Team Modification Flowchart

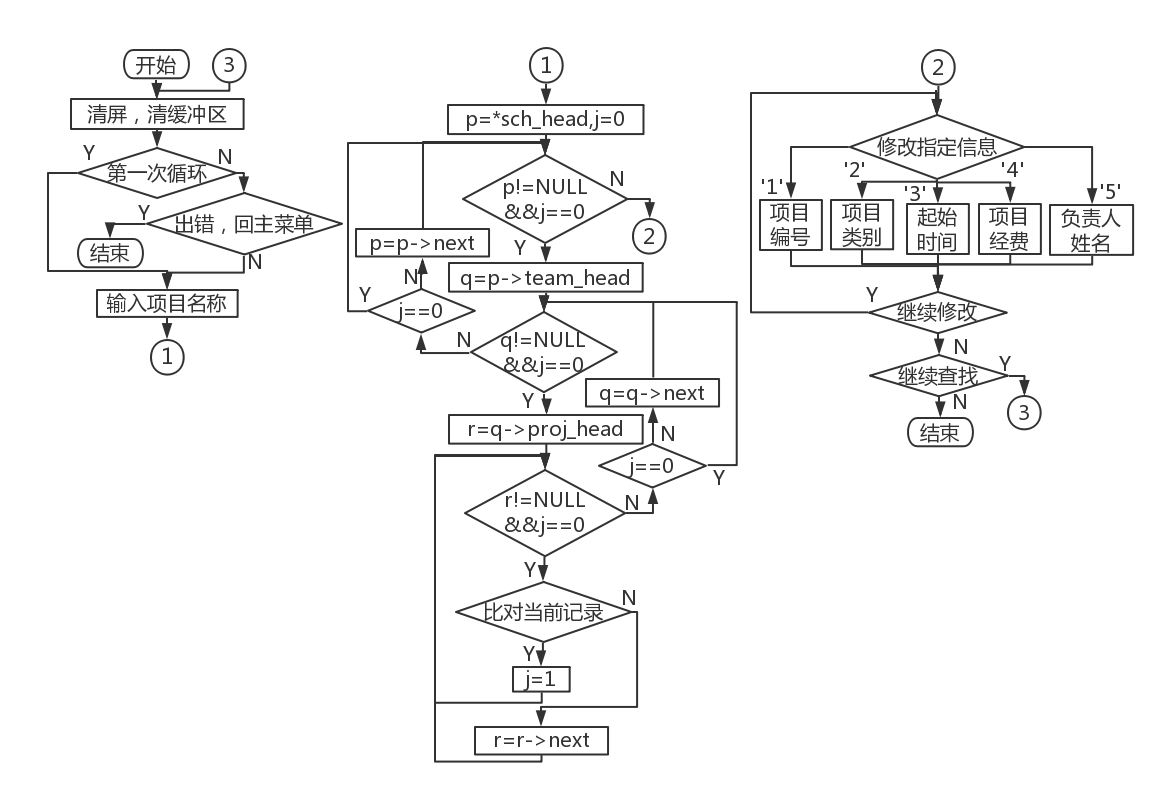


Figure 4-10 \_ Project Modification Flowchart

### 4.2.6 Data modification of scientific research projects

1. Clear screen , clear buffer

2. If it is not the first cycle , enter the error message

3. If you return to the main menu , go to 9, otherwise continue

4. Enter the project number

5. Triple loop matches the current information, if the match is successful, exit the loop and keep the pointer position , otherwise go to 1

6. Select the modification method to modify the specified information

7. If you continue to modify the department information , go to 6, otherwise continue

8. If you continue to modify, go to 1, otherwise continue

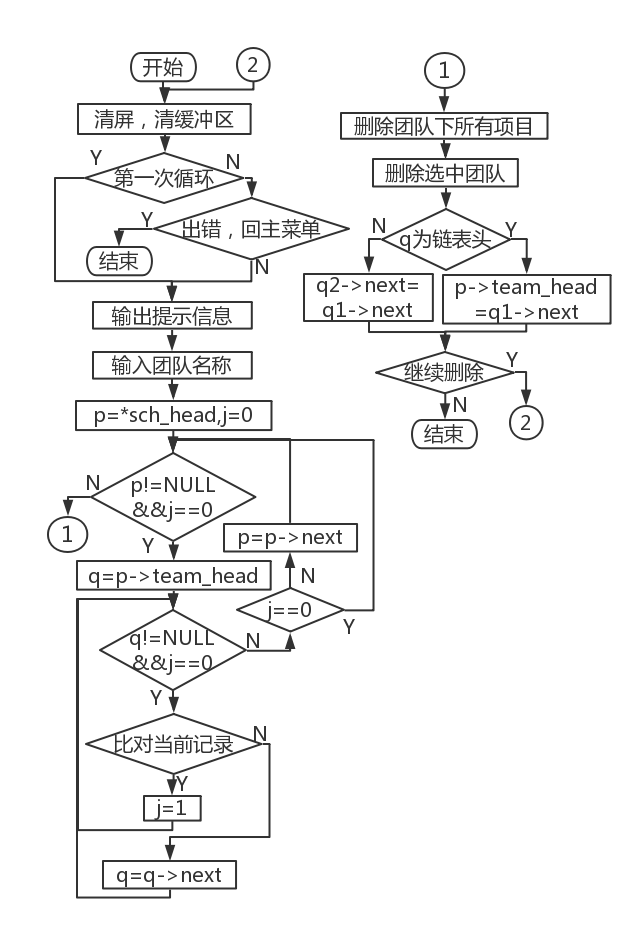
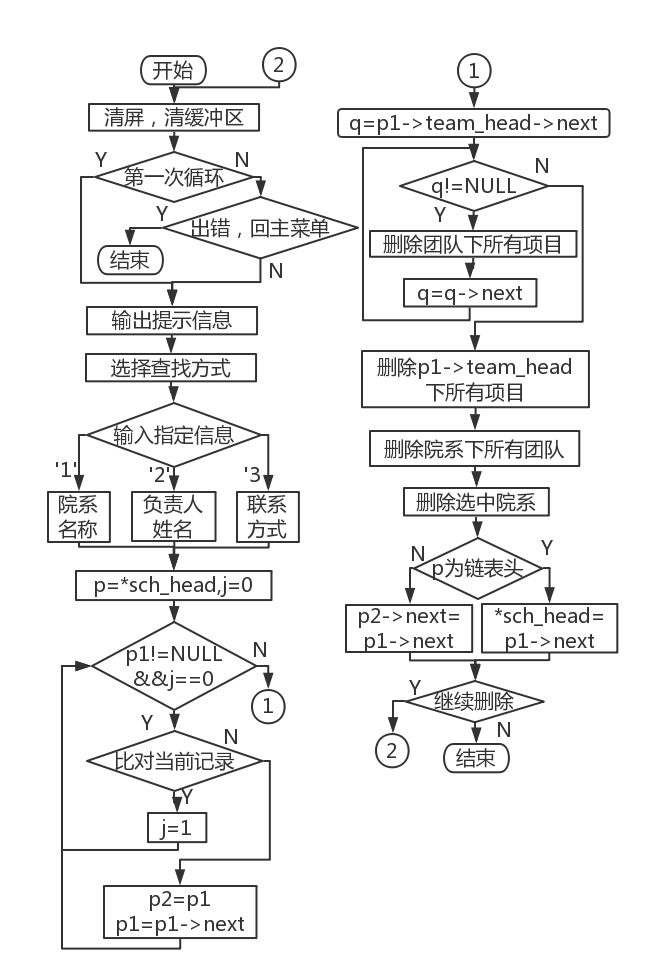
9. end

Figure 4-11 \_ Faculty deletion flow chart Figure 4-12 \_ Team Delete Flowchart

### 4.2. 7 Department data deletion

1. Clear screen , clear buffer

2. If it is not the first cycle , enter the error message

3. If you return to the main menu , go to 15, otherwise continue

4. Choose the way to search for departments

5. Enter match target

6. Loop to match the current information, if the match is successful, exit the loop and keep the pointer position , otherwise go to 1

7. Set q to p1-> team\_head ->next

all project nodes under q

9. p1-> team\_head- >next points to the next node of q , delete q, and q points to p1-> team\_head- >next

10. If p1-> team\_head- >next is empty , continue, otherwise return 9

11. Delete all project nodes under p1-> team\_head

12. Delete p1-> team\_head , p1's team head pointer is empty

13. Delete the current department node

14. If you continue to modify, go to 1, otherwise continue

15. end

### 4.2.8 Data deletion of research team

1. Clear screen , clear buffer

2. If it is not the first cycle , enter the error message

3. If you return to the main menu , go to 10, otherwise continue

4. Choose the way to search for departments

5. Enter match target

6. The double loop matches the current information. If the match is successful, exit the loop and keep the pointer position , otherwise go to 1

all project nodes under q

8. Delete the current team node

9. If you continue to modify, go to 1, otherwise continue

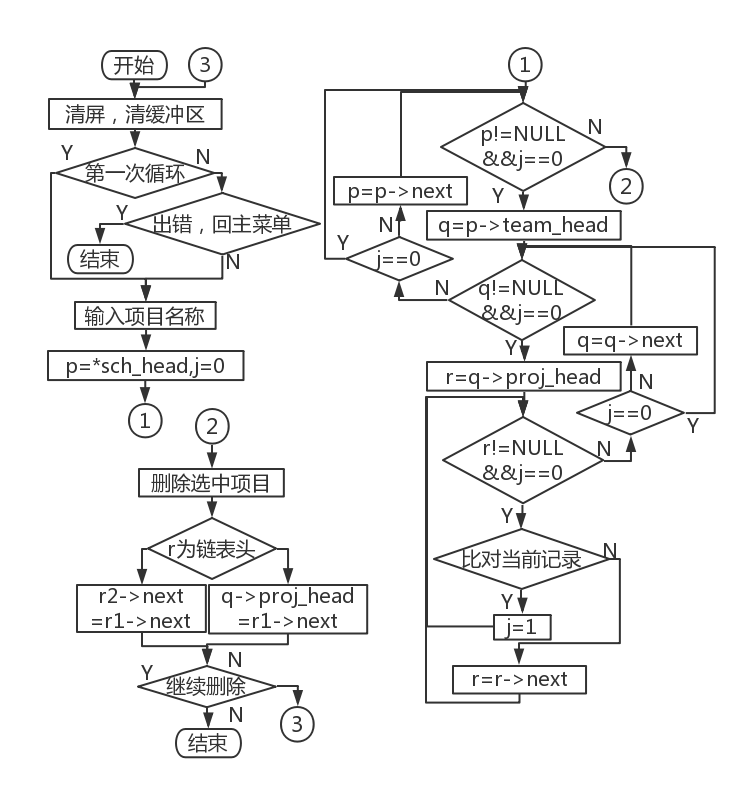
10. end

Figure 4-13 \_ Project deletion flowchart Figure 4-14 Faculty search flowchart

### 4.2.9 Data deletion of scientific research projects

1. Clear screen , clear buffer

2. If it is not the first cycle , enter the error message

3. If you return to the main menu , go to 10, otherwise continue

4. Choose the way to search for departments

5. Enter match target

6. The triple loop matches the current information. If the match is successful, exit the loop and keep the pointer position , otherwise go to 1

all project nodes under q

8. Delete the current team node

9. If you continue to modify, go to 1, otherwise continue

10. end

## 4.3 Information search part

### 4.3.1 Searching for Faculty and Department Information

1. Clear screen , clear buffer

2. If it is not the first cycle , enter the error message

3. If you return to the main menu , go to 9, otherwise continue

4. Choose the way to search for departments

5. Enter match target

6. Loop to match the current information. If the match is successful, print the corresponding department information. At the same time, if you search by the person in charge of the department , exit the loop

7. If the information is not successfully matched , go to 1, otherwise continue

8. If you continue to search, go to 1, otherwise continue

9. end

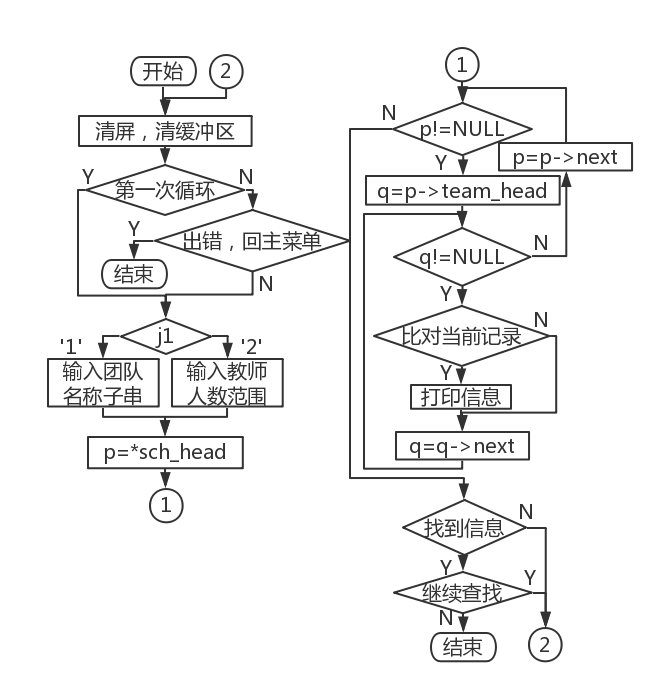


Figure 4-15 team search flowchart Figure 4-16 Project Search Flowchart

### 4.3.2 Research team information search

1. Clear screen , clear buffer

2. If it is not the first cycle , enter the error message

3. If you return to the main menu , go to 9, otherwise continue

4. Choose the way to find the scientific research team

5. Enter match target

6. Circularly match the current information, if the match is successful, print the corresponding department information

7. If the information is not successfully matched , go to 1, otherwise continue

8. If you continue to search, go to 1, otherwise continue

9. end

### 4.3.3 Research project information search

1. Clear screen , clear buffer

2. If it is not the first cycle , enter the error message

3. If you return to the main menu , go to 9, otherwise continue

4. Choose the way to search for scientific research projects

5. Enter match target

6. Circularly match the current information, if the match is successful, print the corresponding department information

7. If the information is not successfully matched , go to 1, otherwise continue

8. If you continue to search, go to 1, otherwise continue

9. end

## 4.4 Information statistics part

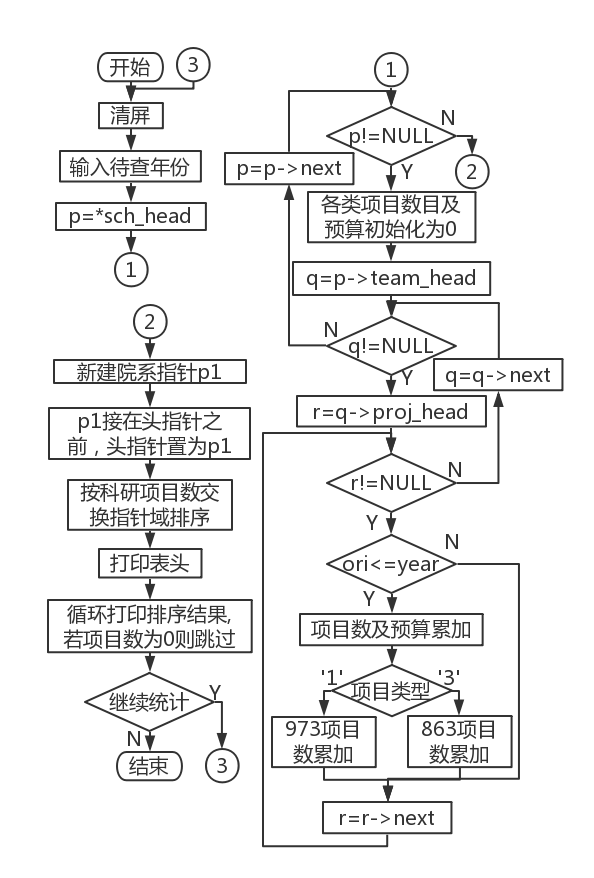
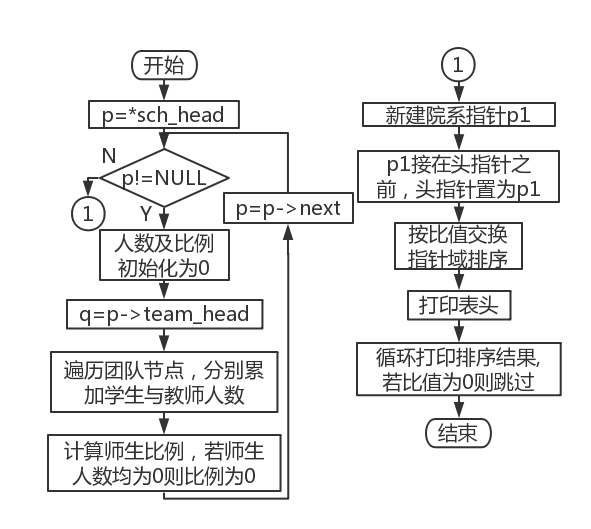


Figure 4-17 Statistical teacher-student ratio flow chart Figure 4-18 Flowchart of Statistical Item Quantity

### 4.4.1 Teacher-student ratio

1. p rewinds the head of the linked list

2. The number of teachers and students and the ratio are initialized to 0

3. At the team level, the number of teachers and students of each team is cyclically summed , and the ratio of teachers and students is calculated at the same time

4. Create a new blank department structure pointing to the original linked list header

5. The head of the linked list points to the newly created node

6. Exchange the pointer field to sort the departments in descending order of the ratio of the number of teachers and students

7. Delete the new node and reset the head pointer

8. Cyclic printing of department information

9. end

### 4.4.2 Number of items

1.p back to the head of the linked list

2. Enter the year to be checked

3. Set q to p-> team\_head

4. At the project level, cyclically count all kinds of project proposals whose starting time is before the year to be checked , and add up the budget at the same time

5.q points to the next team node

6. If q is NULL, p points to the next department node, go to 3, no

7.p is NULL , exit the loop

8. Create a new blank department structure pointing to the original linked list header

9. The head of the linked list points to the newly created node

10. The exchange pointer field sorts the faculties and departments in descending order of the total number of projects

11. Print header

12. Loop printing team information

1 3. Delete the newly created node, reset the head pointer and go to 4

14. end

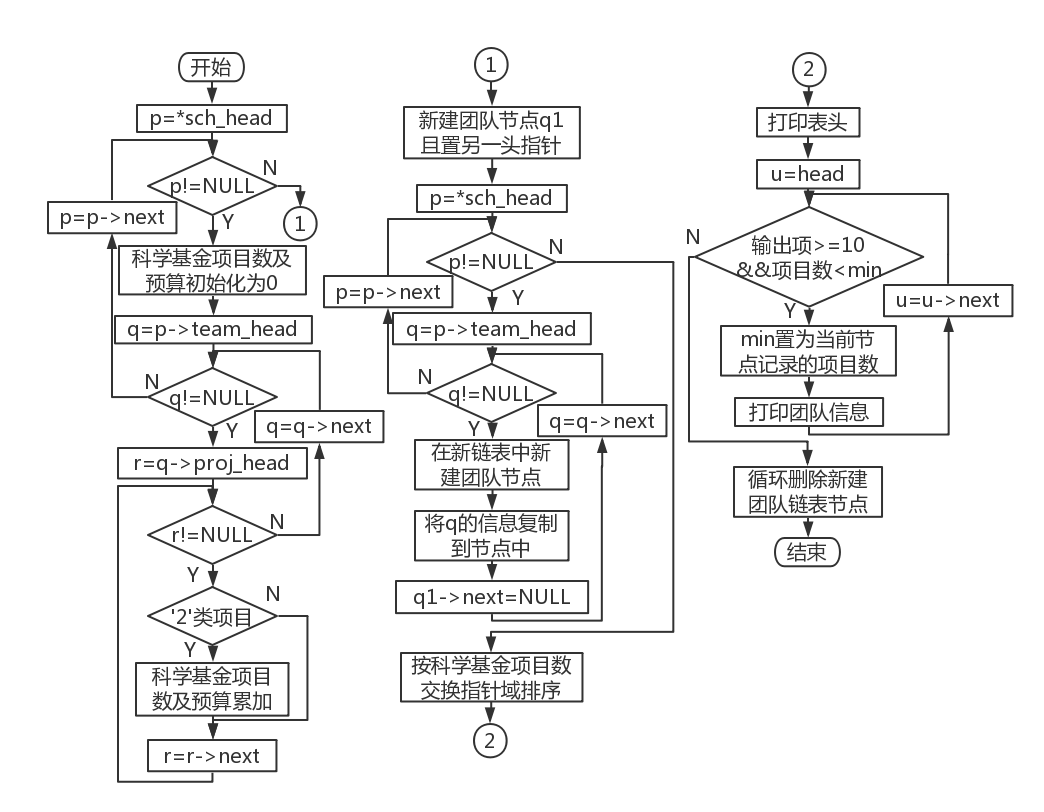


Figure 4-19 Flowchart of counting the number of NSFC projects

### 4.4.3 Number of National Natural Science Foundation of China projects

1. p back to the head of the cross chain

2. Set q to p-> team\_head

3. The number of science fund projects and the corresponding budget are initialized to 0

4. Accumulate the budget and the number of science fund projects at the project level

5. q points to the next team node

6. If q is NULL, p points to the next department node, go to 2, otherwise go to 3

7.p is NULL , exit the loop

8. Create a new independent blank team node as the head of the independent linked list

9. p Rewind the head of the cross chain

10. Set q to p-> team\_head

11. Create a new independent blank team node after the header node

12. Copy the current team information to the team node , set q->next to NULL

13. q points to the next team node

14. If q is NULL, p points to the next department node, go to 10, otherwise go to 11

15.p is NULL , exit the loop

16. The exchange pointer field sorts the teams in the independent linked list in descending order of the number of science fund projects

17. Print team information in a loop , when the number of printed records is greater than 10 and the number of current items is less than the saved minimum value, a loop will pop up

18. Cyclic deletion of new independent team linked list

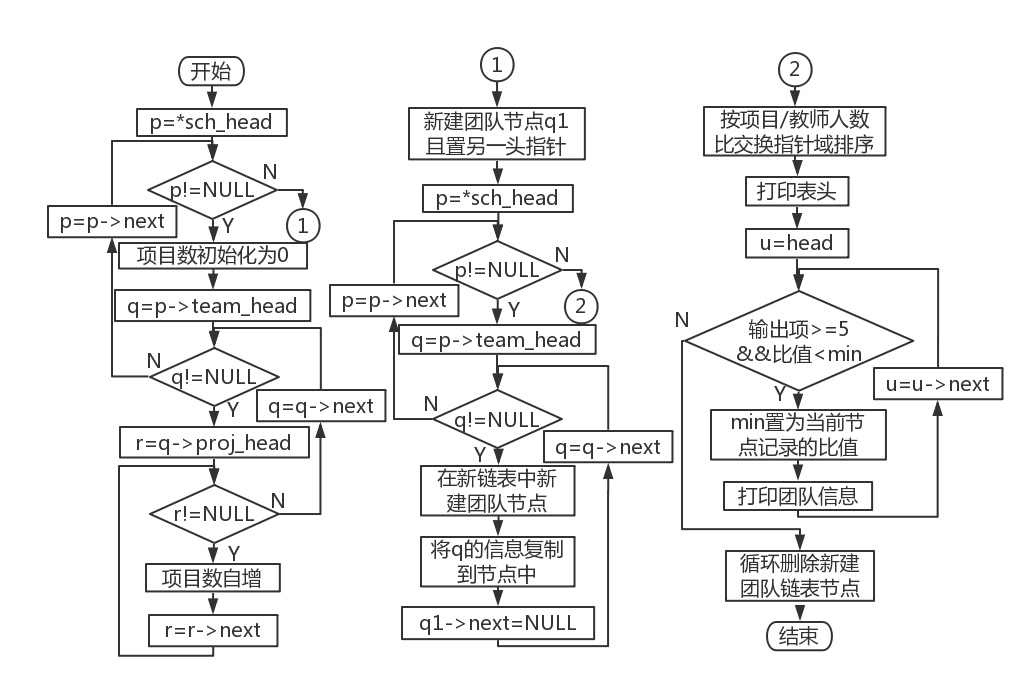
19. end

Figure 4-20 Statistical Item/Teacher Ratio Flowchart

### 4.4.4 The ratio of the number of team projects to the number of teachers

1. p back to the head of the cross chain

2. Set q to p-> team\_head

3. The number of projects under the team is initialized to 0

4. Count the total number of projects under the team at the project level

5. Calculate the project /teacher ratio for the team

6. q points to the next team node

7. If q is NULL, p points to the next department node, go to 2, otherwise go to 3

8.p is NULL , exit the loop

9. Create a new independent blank team node as the head of the independent linked list

10 . p rewind the cross chain watch head

11. Set q to p-> team\_head

12. Create a new independent blank team node after the header node

13. Copy the current team information to the team node , set q->next to NULL

14. q points to the next team node

15. If q is NULL, p points to the next department node, go to 10, otherwise go to 11

16.p is NULL , exit the loop

17. The exchange pointer field sorts the teams in the independent linked list in descending order according to the project/teacher ratio

18. Cyclically print team information , when the number of printed records is greater than 5 and the current ratio is less than the saved minimum value, the loop will pop up

19. Cyclic deletion of new independent team linked list

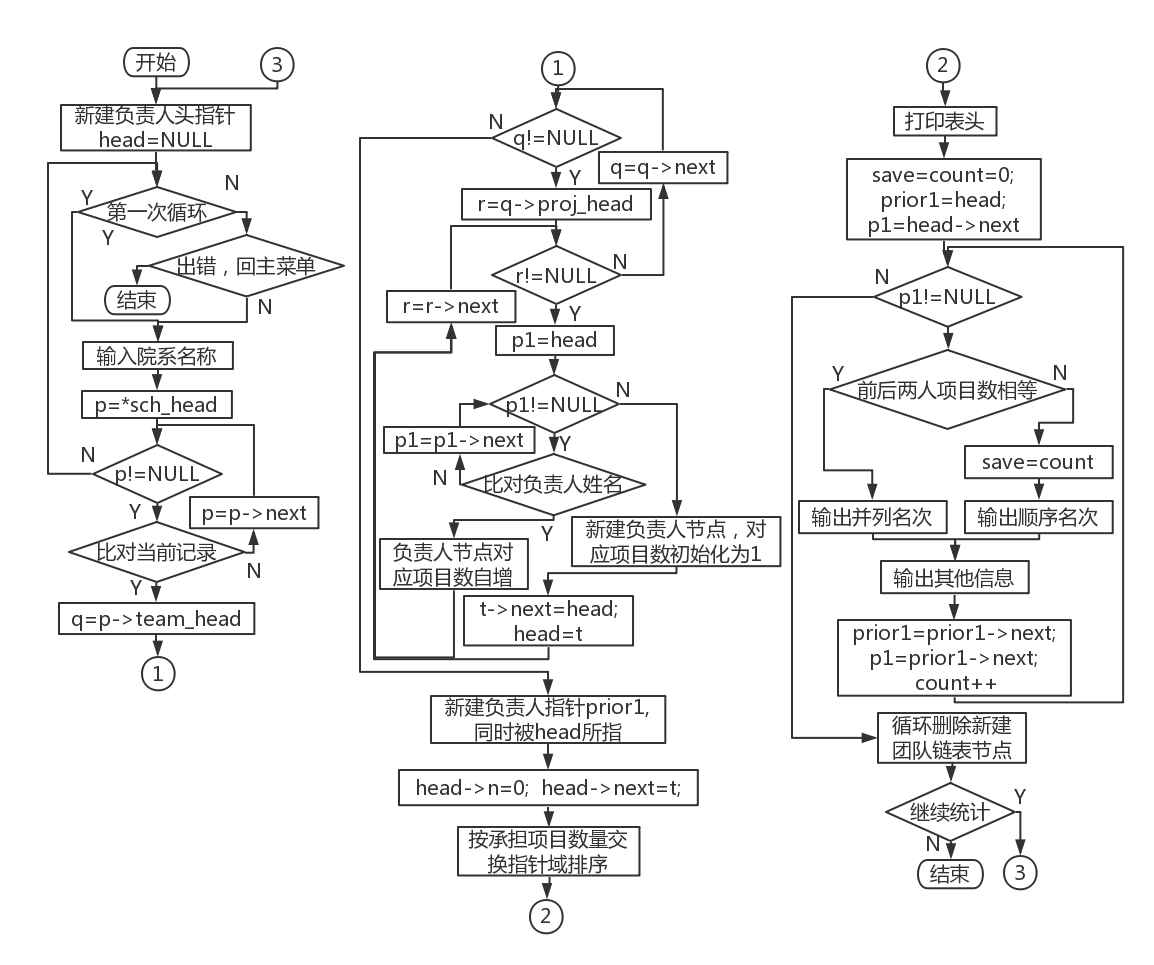
20. end

Figure 4-21 Flowchart of Statistical Project Allocation

### 4.4.5 Number of projects undertaken by the team leader

1. Create a new responsible node head and initialize it to NULL

2. If it is not the first cycle , enter the error message

3. If you return to the main menu , go to 19, otherwise continue

4. Enter the department name

5. Circularly match the name of the department in the department layer , if the match is successful, continue, otherwise go to 1

6. Set q to p-> team\_head

7. r set to q-> proj\_head

8. Traversing the person in charge single-layer linked list, if the names in the two nodes match successfully, the corresponding statistics + 1 ; otherwise, create a new person in charge node to store information about the new person in charge

9.r points to the next project node

10. If r is NULL, q points to the next team node, go to 7, otherwise go to 8

11.p is NULL , exit the loop

12. Create a new blank person in charge structure pointing to the head of the original linked list

13. The head of the linked list points to the new node

14. The exchange pointer field sorts the faculties and departments in descending order of the number of projects undertaken

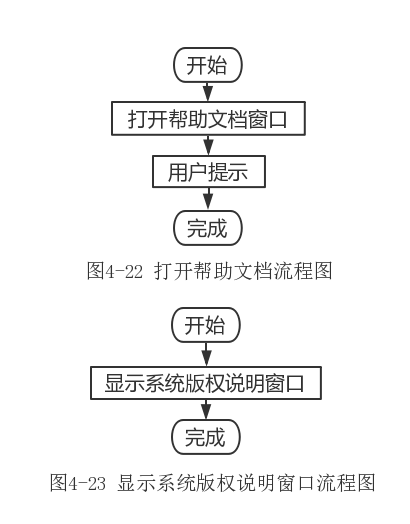
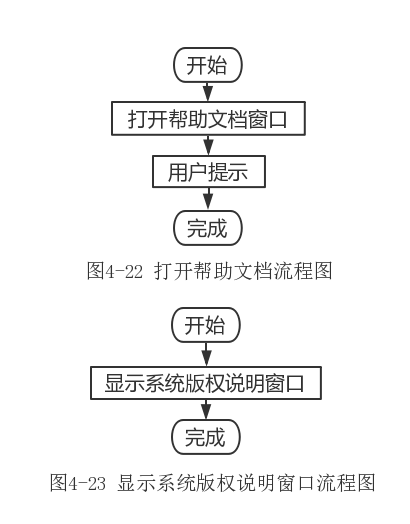
15. Print header

the information of the person in charge . If the number of projects undertaken by the two people before and after is the same , their rankings will be tied

17. Cyclic deletion and new independent person in charge linked list

18. If you continue to search, go to 1, otherwise continue

19. end

## 4.5 Help section

### 4.5.1 Help topics

1. Open the help document window 2. End

### 4.5.2 System Copyright Description Theme

1. Display copyright statement window 2. End

# 6. Running test and result analysis

## 6.1 Interface and window operation

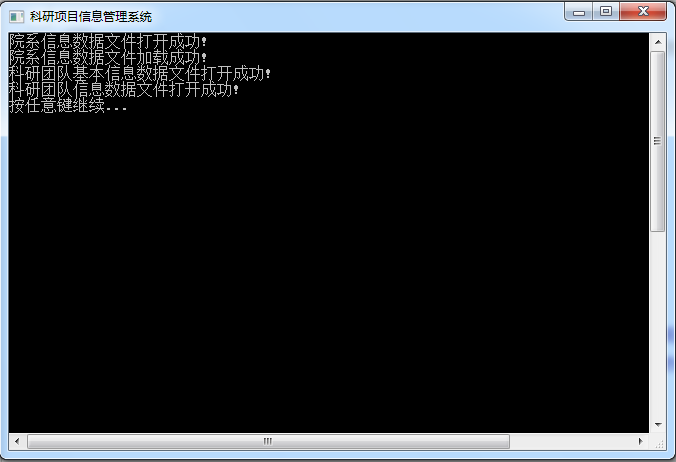


Figure 6-1 Data loading interface



Figure 6-2 System initial interface and default interface



Figure 6-3 \_ exit confirmation interface

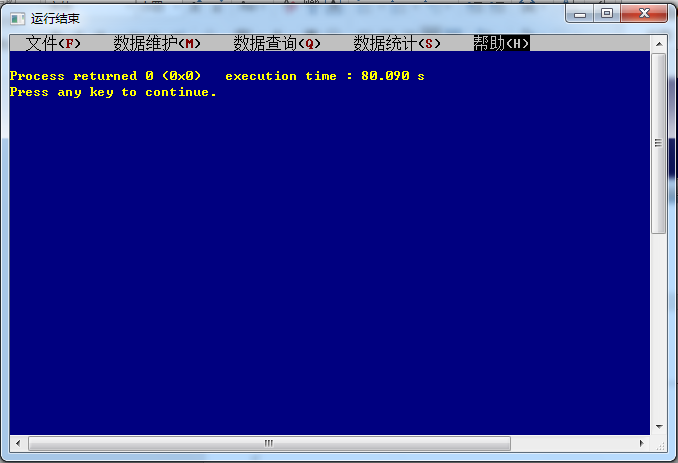


Figure 6-4 \_ exit interface

## 6.2 Operation of data maintenance function



Figure 6-5 Faculty / Research Team/Research Project Information Maintenance Directory Interface

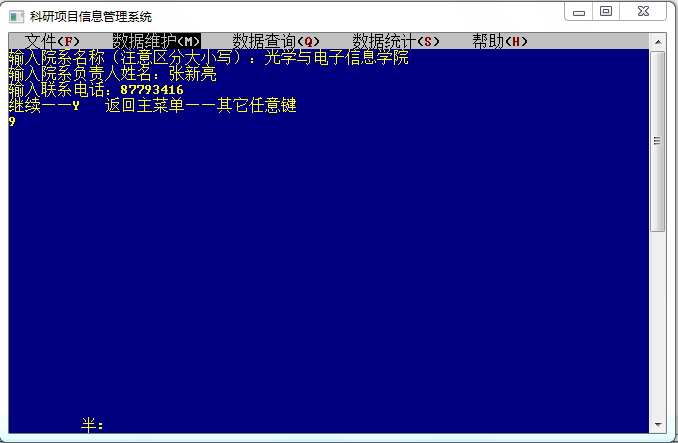


Figure 6-6 Create a new department interface



Figure 6-7 Create a duplicate department interface

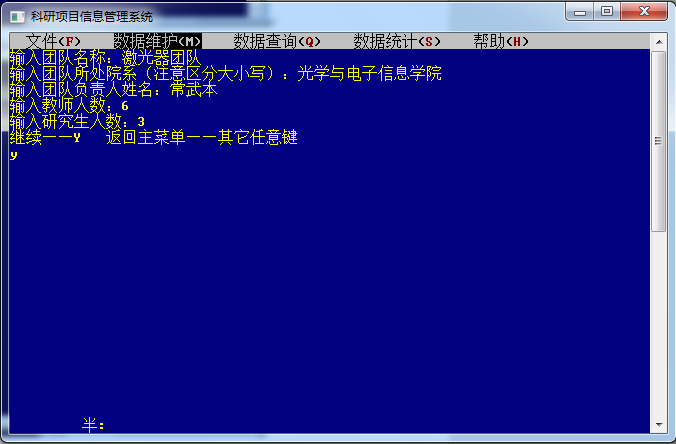


Figure 6-8 Create a new team interface

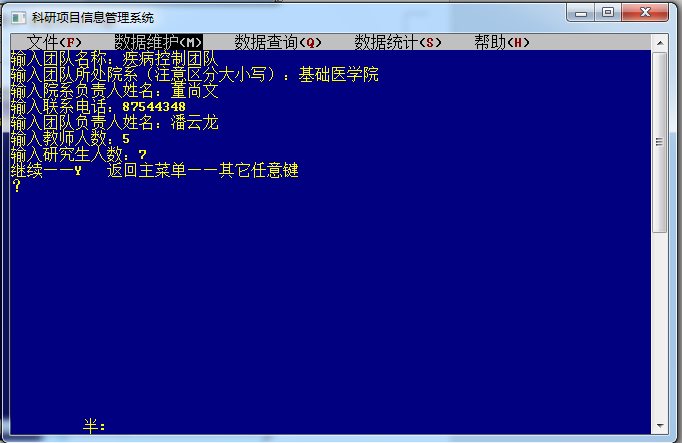


Figure 6-9 Create a team interface under a new department



Figure 6-10 Create a duplicate team interface

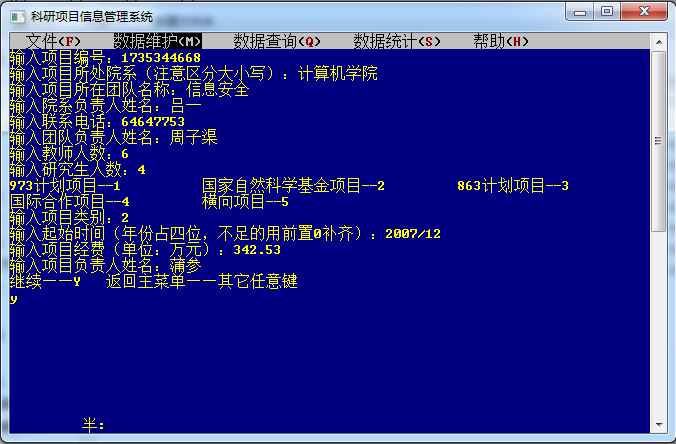


Figure 6-11 The project interface under the newly created new department

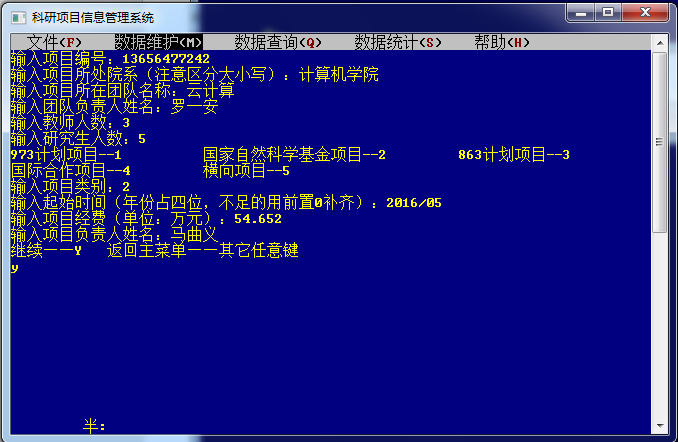


Figure 6-12 Create a new team under the project interface

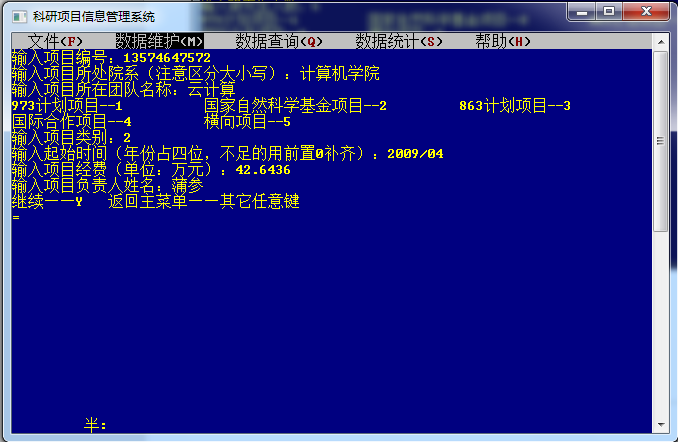


Figure 6-13 Create a new project interface



Figure 6-14 project creation repeat interface

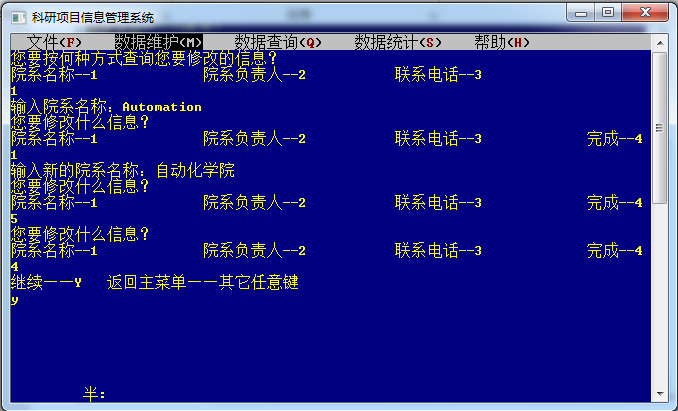


Figure 6-15 Query and modify the department information interface by department name



Figure 6-16 No department to be modified interface found

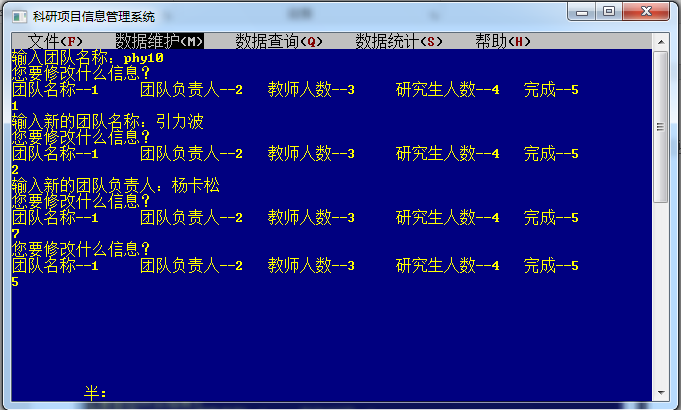


Figure 6-17 Find the team and modify the team information interface



Figure 6-18 No team interface to be modified was found



Figure 6-19 Find and modify project information interface



Figure 6-20 No project to be modified interface found



Figure 6-21 Find and delete the department interface by the person in charge of the department

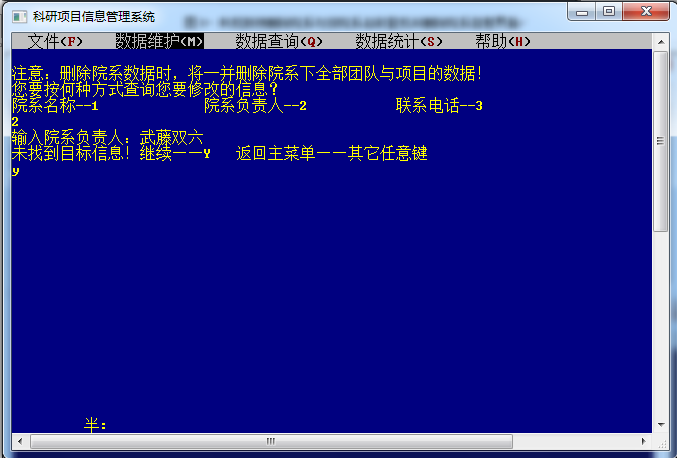


Figure 6-22 No department to be deleted interface found

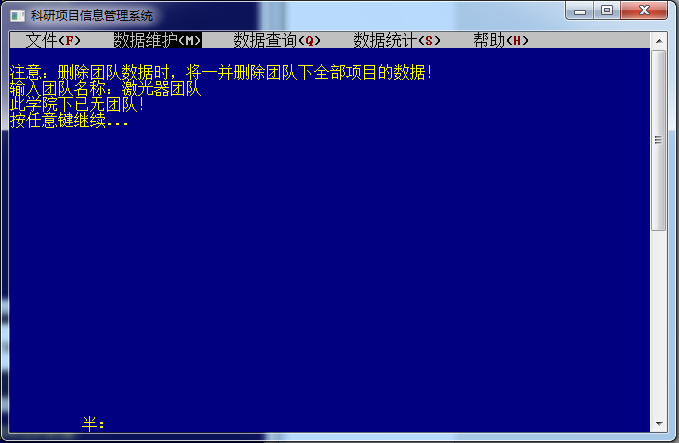


Figure 6-23 Delete the only team interface under the department

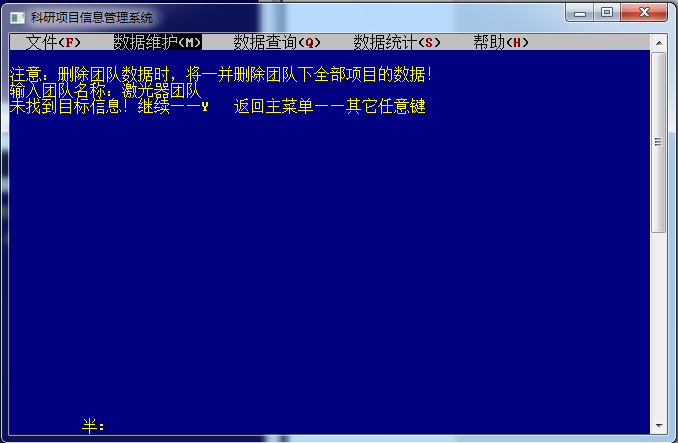


Figure 6-24 No team to be deleted interface found



Figure 6-25 No item to be deleted interface found



Figure 6-26 Delete the only project interface under the team

## 6.3 Operation of data query function



Figure 6-27 Query the department information interface by the person in charge of the department



Figure 6-28 Query the department information interface by part of the department name



Figure 6 -29 No corresponding department interface found



Figure 6-30 Query the scientific research team information interface by part of the team name



Figure 6-31 Query the scientific research team information interface according to the number of teachers



Figure 6 -32 No corresponding research team interface found

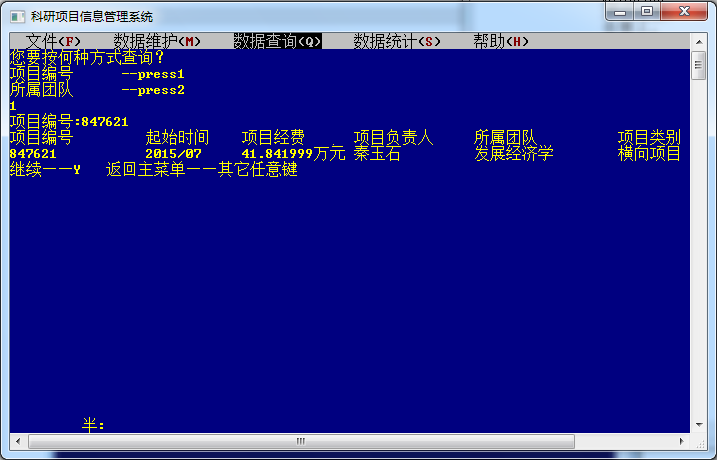


Figure 6-33 Query scientific research project information interface by project number



Figure 6-34 Query the scientific research project information interface by the team to which the project belongs



Figure 6 -35 No corresponding scientific research project interface found

## 6.4 Operation of data statistics function

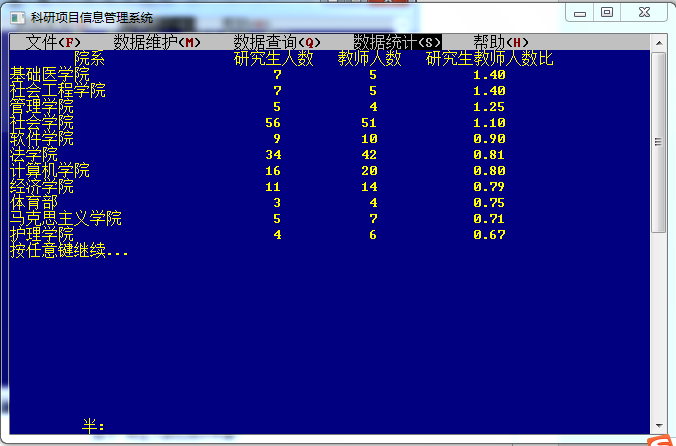


Figure 6-36 Teacher-student ratio statistics interface



Figure 6-37 Project Quantity Statistics Interface



Figure 6 -38 National Natural Science Foundation of China project statistics interface



Figure 6-39 The ratio interface between the number of team projects and the number of teachers



Figure 6 - 40 team leaders undertake the number of projects interface

## 6. System information function running status



Figure 6-41 help document execution interface

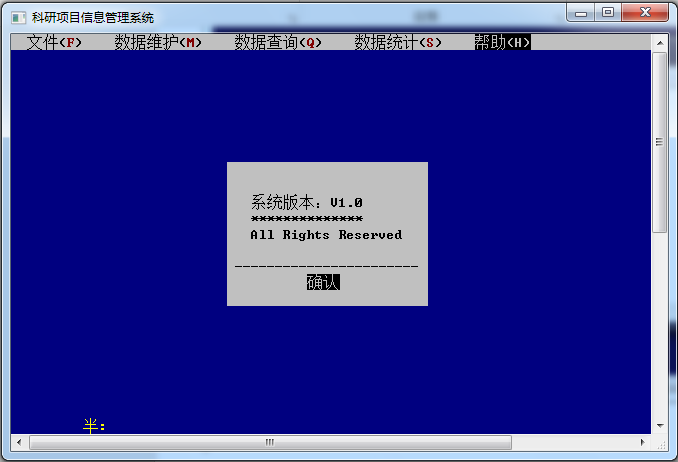


Figure 6-42 System-related window execution interface

# 7. Summary

The first difficulty I encountered this time was the problem of data organization structure . Since I haven't touched the C language for nearly a month , I have forgotten the knowledge related to the linked list to a certain extent, and the cross-linked list part needs to be learned by myself . Fortunately, the C language foundation I played last semester is relatively solid, so after scanning the book, the program structure part is not a problem .

The second is the problem of file data transfer. Similarly , my understanding of file input and output last semester stayed at a very elementary level , but I wanted to simplify the operation of file input and output as much as possible. Combined with the data structure, I must use the two functions fread and fwrite . Reading and writing styles are also reviewed. This shows that people's memory ability is limited after all . If you want to master and maintain efficient programming ability, you need a programmer to pile up a lot of code operations on a daily basis. On the one hand, this is to keep memorizing some hard grammar knowledge , and on the other hand, it also helps to maintain the programming feel . At the same time, as a programmer , you must have a strong self-learning ability, not only to quickly learn new knowledge that you have never been exposed to before in the face of unfamiliar tasks, but also to keep up with the ever-changing computer field.

Surprisingly, the program debugging that I expected to be stuck for a long time did not create too many obstacles for me in the end. I think it is not only because I repeatedly compile to 0 errors and 0 warnings every time I write a function , the habit of compiling errors rarely occurs later, but more importantly, I will repeatedly confirm its contents in my mind or even on paper when I write a function . Such a cautious behavior enabled me to pass the program debugging process after solving a few logic errors . But on the other hand , confirming the execution status over and over again is also a manifestation of my inability to abstract a piece of code smoothly , which reduces my programming efficiency to a certain extent. So then again , only a large amount of daily code stacking can be exchanged for high-quality and efficient programming in tasks .

Having said that , the link that cannot be saved is still not saved at all-data testing. In order to ensure the stability of program operation , a large amount of data testing is inevitable. Specifically for this course design , in order to ensure the diversity of the linked list structure, it is necessary to have zero, one, or multiple teams affiliated to the department, and these situations include zero, one or more projects affiliated respectively. This kind of diversity requires a large amount of basic data as a basis ; when maintaining data , I have to consider the situation of establishing departments, teams and projects at the same time in extreme cases , or a department with all its teams and projects In the case of deleting together; in data statistics, in order to widen the gap between the top ten and top five values and deliberately make the juxtaposition appear, I also need to pay attention to everything when entering data, so in the end I almost spent on data testing and program debugging . for the same amount of time.

the written function body with the given text interface at the end. It took me two days to understand the sample code of the text interface . When I improved the interface later, I was able to understand the functions of most of the code and make changes according to individual needs ; while splicing, I also gradually produced some The idea of increasing user-friendliness and enriching the system's functions, so I also came into contact with some new functions . In this process, program debugging really takes up a lot of links, some are due to omissions, and some are indeed caused by lack of knowledge: such as the inexplicable failure of mouse input after the execution of the sys tem ( "pause" ) statement I don't understand it at all, but before I know this feature, I can only guess that I made a misoperation on the mouse input related functions before, so I basically checked all the text interface related functions again.

At the same time , this system considers user-friendliness as much as possible within the scope of its ability. For example, when wrong information is entered, the system will actively prompt and force re-input; try to allow users to input through dialog boxes instead of keyboards to reduce the possibility of user misoperations ; Go to the main menu to prevent the pointer from being confused or the program from falling into an infinite loop; in as many links as possible, the user is given the function of continuing to operate or returning to the main menu.

# 8. References

[1] Cao Jichang, Lu Ping, Li Kai. C Language Programming, Beijing: Science Press , 2008

[2] [Zhang Yin](http://www.amazon.cn/mn/searchApp?searchWord=%E5%BC%A0%E5%BC%95) . C programming basic course design, Hangzhou: Zhejiang University Press, 2007

[3] [Huang Ming](http://www.amazon.cn/mn/searchApp?searchWord=%E9%BB%84%E6%98%8E) , [Liang Xu](http://www.amazon.cn/mn/searchApp?searchWord=%E6%A2%81%E6%97%AD) , [Wan Hongli](http://www.amazon.cn/mn/searchApp?searchWord=%E4%B8%87%E6%B4%AA%E8%8E%89) . C language course design, Beijing: Electronic Industry Press , 2006