# Application of linear list Experiment Report

Class:计科201 student id:19404150205 name:刘志扬

Student id:20401010102 name:毛威任

Student id:19402010318 name:张俊杰

### Experiment purpose

1、Use the basic operations to implement the specific operations for the linear table;

2、Master the application of file operations;

3、Improve the understanding of the data structure of linked storage structure, and gradually cultivate the programming ability to solve practical problems.

### 2.Experimental environment

A computer with Visual C ++ 6.0 / CFree.

This experiment has 4 class hours in all.

### 3.Experiment content

Design a classmate's address list, requested as follows:

 Each student in the address list contains the following information: student id、name、telephone number. If you need more fields, please add them yourself.

 The program has a main menu containing the following functions:

（1） Add a record: Add a student record from the input.

（2） Delete a record: Delete a student record according to the student id from the input.

（3） Output all records: Display all the records in the address list.

（4） Search by name: Input the student name and then output the whole information of the student.

（5） Save records: Save all the records in the address list to a certain file.

（6） Clear records: Delete all the records in the address list and then delete the file.

（7） Quit

hint：

 When the program starts, it should be determined whether there is a record file. If the file exists, read each record from it to the list.

 After the user selects and completes a function of the main menu, the program should return to the main menu.

 When a record is added, it should be inserted into the tail of the list.

 If a record does not exist when performing delete or and search operation, the program should output some information to the user.

 You do not need to write files when adding records or deleting records.

 When you want to save a record you’d better overwrite the file. (Or delete the original file first, and then save all the records)

 Each module is written in the form of a function, called by the main function.

optional：

 Add a sorting function in the main menu, the sorting result should be in an ascending order according to the student number. Sorting methods can be done by bubble sort or insert sort.

### 4.Important data structures

Using the simple Simple Array Implementation of List.

#pragma once

#include<string>

using namespace std;

typedef struct seqlist //顺序表结构体

{

string name[500];

string ID[500];

string phone[500];

int count;

}seqlist;

seqlist\* p\_list = new seqlist;

### 5.implementation analysis

Using while loop to make the menu. than pack each demand into function to be called into the menu loop.

Menu loop:

while (1)

{

if (quit == true)

break;

printmenu();

int express = \_getch();

system("cls");

switch (express)

{

case'1':

{

string temp1,temp2,temp3;

print11();

cin >> temp1;

print12();

cin >> temp2;

print13();

cin >> temp3;

addrecord(temp1, temp2, temp3, p\_list);

cout << "新数据已添加" << endl;

break;

}

case'2':

{

string temp;

print2();

cin >> temp;

deleteID(temp, p\_list);

break;

}

case'3':

{

printlist(p\_list);

cout << "按任意键继续" << endl;

express = \_getch();

break;

}

case'4':

{

print4();

string temp;

cin >> temp;

searchname(temp, p\_list);

break;

}

case'5':

{

writetext("1.txt", p\_list);

break;

}

case'6':

{

deletefile(p\_list);

break;

}

case'7':

{

quit = true;

break;

}

}

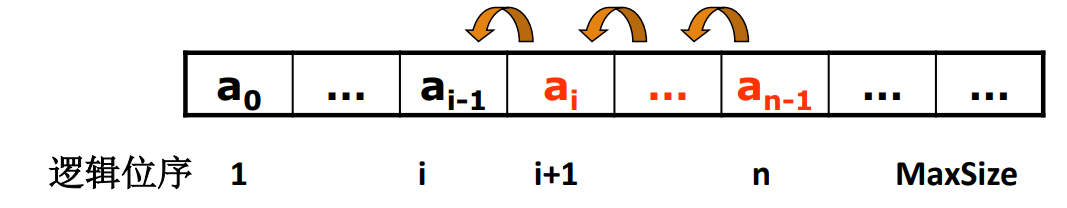
}

**Search function:**

To search each elements in the data struct

**Delete function:**

Using the next element cover the previous element



**Add new element function:**

Using the previous element cover the next element than put the new element into the space that was vacant.

### 6.debugging problem analysis

Problem: When running the file read function there was a problem that the data struct can’t save the data which read form the txt file.

Solution: we find that the memory space of the data struct wasn’t be given.

So we give it the dynamic memory.

seqlist\* p\_list = new seqlist;

### 7.summary

Through this experiment, we learned how to save data with the array list data struct and also reviewed how to read data and write data with txt flies.

### 8.crew division

|  |  |  |
| --- | --- | --- |
| Group division | | |
| Member name | Work done | Completion situation |
| 刘志扬 | Data struct using and each demand function | complete |
| 毛威任 | File read and write | complete |
| 张俊杰 | Data struct using and each demand function | Complete |