address list design **Experiment Report**

Class: Student ID 1: 2040110113 Name 1: 郭旭 Experiment Date:2021/11/2

Student ID 2: 20401010116 Name 2:黄柏宁

Student ID 3: 20401010112 Name 3:俞幸宏

1. **Experimental 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.

1. **Experimental environment**

A computer with Visual C ++ 6.0 / CFree.

This experiment has 4 class hours in all.

1. **Experimental content**

### address list design

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.

1. **Important data structures**

**typedef struct Address**

**void showMune()**

**void creatList(list\*& l)**

**void addList(list\*& l)**

**int deleteList(list\*& l, int a)**

**void output(list\*& l)**

**int search1(list\* l,char\* name1)**

**int search2(list\* l, int a)**

**void store(list\*& l)**

1. **Implementation analysis**

**首先我们用结构体定义每个基本的数据单位，用来存放本身数据和下一个节点的位置，并且用一个头节点来用于找到一整组数据的地址。其中有创建函数，删除函数，添加函数，输出函数，搜索函数来进行对单链表的操作。对应创建数据组，删除数据，添加数据，输出数据和搜索节点的功能，同时定义了输出菜单和保存功能，来实现用户交互和功能，在菜单中用户可以通过输入对应数字来进行相应操作，并且可以将数据存到文件这一操作是通过文件流来进行实现的，总体可以实现相关功能。**

1. **Debugging problem analysis**

**调试过程中具体可能还是遇到线性表操作不熟悉的问题，比如指针地址的使用不熟练导致传入参数不匹配等问题，还有搜索时next\*指针没有更换的问题，但是通过调试都可以解决。有个较大问题时单链表在进行搜索时的时长问题，他在对较多数据进行处理的时候时间长度会很长，尽管删除和插入操作很方便，但是还是需要进行搜索的操作。在这种处理这种很多数据的情况下，可能还是用数据表会好一些。**

1. **Summary**

**熟悉了使用基本操作实施线性表的具体操作掌握文件操作的应用，提高了对链接存储结构数据结构的认识，培养了解决实际问题的编程能力。用线性表将系统进行搭建并且熟练了线性表的操作**

1. **Crew Division**

|  |  |  |
| --- | --- | --- |
| **Group division** | | |
| **Member name** | **Work done** | **Completion situation** |
| **郭旭** | **代码编写实验报告编写** | **Done** |
| **黄柏宁** | **代码编写实验报告编写** | **Done** |
| **俞幸宏** | **代码编写实验报告编写** | **Done** |