

重庆交通大学信息科学与工程学院

综合性设计性实验报告

题 目	通讯录管理系统
--------	---------

课程名称	面向对象程序设计 A
------	------------

专业班级	计算机类 21 级曙光班
------	--------------

学 号	632107060506
--------	--------------

姓 名	李幸洋
--------	-----

指导教师	李益才
------	-----

2022 年 5 月

综合性设计性实验评分标准

项目	评分标准与等级	等级	得分
演示与报告质量	A、程序演示完全正确，界面美观，能正确回答老师问题；报告规范，分析清楚，严格按照要求条目书写，阐述清楚。能针对教师所出的针对特定问题的综合性设计性题目利用面向对象程序设计思想进行分析和设计合适的解决方案，并能使用合适的编程平台进行编程、调试、测试和实现。		
	B、按要求完成 90%及以上功能，界面尚可，能正确回答老师问题；报告规范，分析清楚，个别条目书写不完全符合要求，阐述基本清楚。能针对教师所出的针对特定问题的综合性设计性题目利用面向对象程序设计思想进行分析和设计较合适的解决方案，并能使用合适的编程平台进行编程、调试、测试和实现。		
	C、按要求完成 80%及以上功能，基本能回答老师问题；报告基本规范，分析基本清楚，存在 20%以内条目书写不完全符合要求。在教师的指导下，能针对教师所出的针对特定问题的综合性设计性题目利用面向对象程序设计思想进行分析和设计较合适的解决方案，并能使用合适的编程平台进行编程、调试、测试和实现，成功调试功能达 80%以上。		
	D、按要求完成 70%及以上功能，能回答老师多数问题；报告基本规范，存在 30%以内条目书写不完全符合要求。在教师的指导下，能针对教师所出的针对特定问题的综合性设计性题目利用面向对象程序设计思想进行分析和设计较合适的解决方案，并能使用合适的编程平台进行编程、调试、测试和实现，成功调试功能达 70%以上。		
	E、存在 30%以上功能未完成或抄袭；报告不规范或存在 30%以上条目书写不完全符合要求。无法采用正确的方法完成项目分析和解决方案设计或成功调试测试功能超过 30%未完成。		
指导教师：		得分	

目 录

综合性设计性实验评分标准.....	2
1. 设计需求概述.....	1
2. 系统总体设计.....	1
2.1 系统功能设计	1
2.2 数据模型设计	2
3. 系统实现	5
3.1 系统菜单实现	5
3.2 系统功能实现	6
4. 测试及测试结果	15
功能测试	15
集成测试	19

1. 设计需求概述

通讯录管理系统应用于日常生活以及工作，要能够帮助人们记忆一些人的信息，同时也需要能够对这些人的信息进行管理，第一要能够保存若干联系人的信息、第二要能够按名字或电话快速的查找到联系人、第三要能够对联系人进行管理，例如添加、删除、修改等。最后用户要能够快速的上手使用，也就是该系统要简单易懂且使用效率高。

2. 系统总体设计

2.1 系统功能设计

- 查看联系人

该功能能够查看所有联系人，用户选择该功能后需要直接呈现所有信息，也就是能够看到所有联系人的信息。

- 添加联系人

用户选择该功能后，根据提示信息依次把要添加的联系人的信息输入，系统根据这些信息创建一个信息类的对象并把该对象加入容器中进行管理。

- 删除联系人

此功能由用户选择后，先打印出所有联系人，如果没有联系人即退出该功能，随后给出提示用户选择待删除联系人编号，系统自动删除对应编号联系人，最后系统会给出删除成功或失败提示信息。

- 查找联系人

此功能可以由用户选择通过电话或是姓名查找，通过提示信息输入对应的信息，随后在界面中打印出所有联系人信息。

- 修改联系人

此功能由用户后，系统打印出所有联系人，随后提示用户输入对应联系人编号进行删除，最后系统会给出修改成功或失败提示信息。

- 清空联系人

该项功能比较简单，用户选择后，系统给出是否确认清空，随后系统完成对应操作并给出提示信息。

2.2 数据模型设计

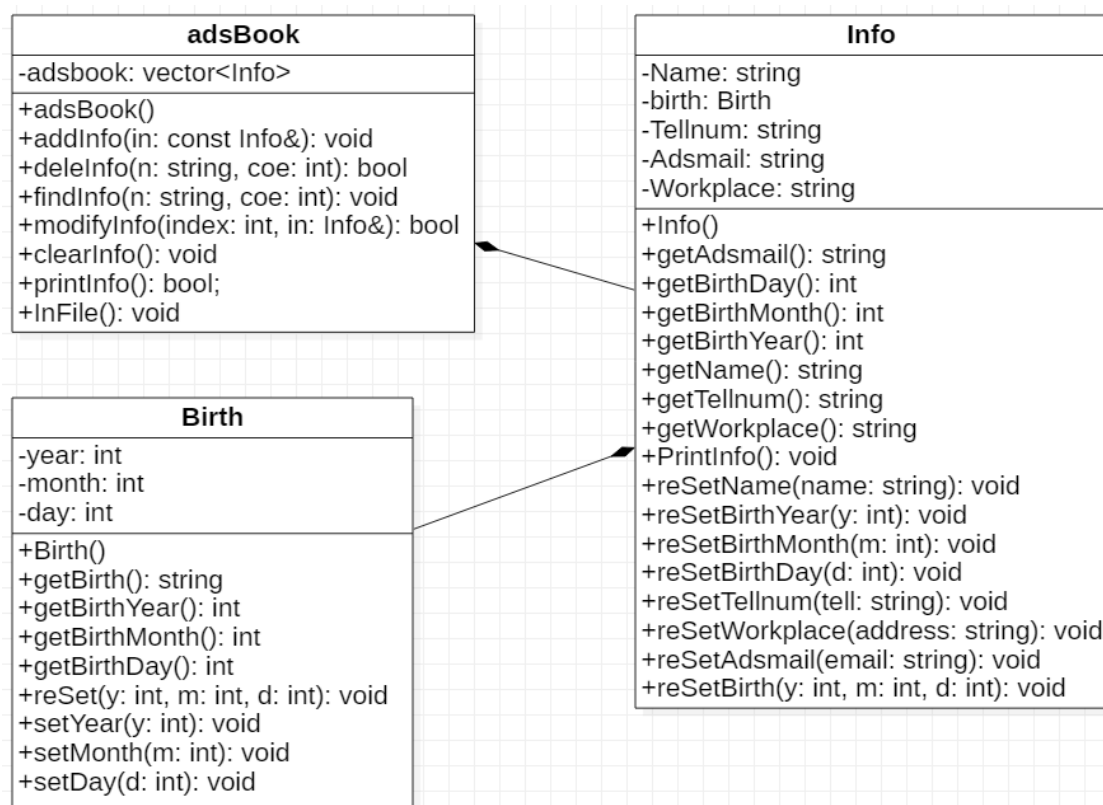
2.2.1 类结构设计

从现实中来看，人与人之间通讯无非是同电话、地址、邮箱等，再加上人的信息，例如，生日、姓名等，就能得到该系统中最基本的一些数据成员。经过更加细致的划分，出生日期与其他不同做一个日期类进行封装，其余均使用 `string` 数据类型，再用一个信息类对上诉的日期类，姓名等信息加以封装。

得到数据成员后，对类中数据的操作方法就是成员函数，其中有两种操作方法操作相同。如修改数据函数、查看数据函数，除此之外还有一些成员函数。

最后，设计一个通讯录类对信息类进行管理，使用 `vector` 容器存取信息类对象，以及提供一些对信息操作的一些方法，也就是成员函数。

下面给出了类模型图，能够看到数据成员（属性名、属性类型），成员函数（返回类型、成员函数名、形参表）。



类模型图

2.2.2 类的实现

类使用了两个文件进行编写，对应上面的类结构图，接下来给出类的声明日期类：

```
1. class Birth
2. {
3.     int year;//出生年
4.     int month;//出生月
5.     int day;//出生日
6. public:
7.
8.     Birth(int , int , int );//初始化出生年、月、日
9.     string getBirth();//使 int 转到 string 型，再返回
10.
11.     //修改或设置年、月、日
12.     void setYear(int);
13.     void setMonth(int);
14.     void setDay(int);
15.     void reSet(int, int, int);
16.
17.     //返回 int 型的年、月、日
18.     int getBirthYear();
19.     int getBirthMonth();
20.     int getBirthDay();
21.
22. };
```

信息类:

```
1. class Info
2. {
3.     Birth birth;//日期
4.     string Name;//姓名
5.     string Tellnum;//电话号码
6.     string Workplace;//工作单位
7.     string Adsmail;//邮箱
8.
9. public:
10.     //初始化数据
11.     Info(string, int , int , int , string, string, string);
12.
13.     //修改或设置数据
14.     void reSetName(string);
15.     void reSetBirthYear(int);
16.     void reSetBirthMonth(int);
17.     void reSetBirthDay(int);
18.     void reSetTellnum(string);
19.     void reSetWorkplace(string);
```

```
20. void reSetAdsmail(string);
21. void reSetBirth(int y, int m, int d);
22.
23. //返回对应类型的数据
24. string getName();
25. string getTellnum();
26. string getWorkplace();
27. string getAdsmail();
28. int getBirthYear();
29. int getBirthMonth();
30. int getBirthDay();
31.
32. //打印信息
33. void PrintInfo();
34. };
```

通讯录类:

```
1. //定义通讯录以及管理方法
2. class adsBook
3. {
4.     vector<Info>book; //vector 存取信息
5.
6. public:
7.     //初始化数据，从数据文本文件中读取
8.     adsBook();
9.
10.    //修改容器中信息
11.    void addInfo(const Info& in);
12.    void addInfo(string, int, int, int, string, string, string);
13.    bool deleInfo(string, int);
14.    void findInfo(string, int);
15.    bool modifyInfo(int, Info&);
16.    void clearInfo();
17.
18.    //打印所有信息
19.    bool printInfo();
20.
21.    //数据写入文件
22.    void inFile();
23. };
```

3. 系统实现

3.1 系统菜单实现

系统菜单为功能选择界面，需要在最初的界面以及每次完成一个功能之后重新呈现，此处采取 while 循环+switch 选择结构完成，每个条件对应一种操作，把这些操作各自模块化为函数，这些函数分另一个文件实现，主函数的架构基本就是如此。

代码如下：

```
1. int main()
2. {
3.
4.     int coe;//选择输入
5.     adsBook adsbook;//通讯录管理对象
6.
7.     while (true)//界面保持
8.     {
9.
10.        mainFace();//主功能界面
11.
12.        cout << "请选择功能: "; cin >> coe;//选择功能操作
13.
14.        //对应功能调用相关函数
15.        switch (coe)
16.        {
17.            case 0:Exitprogram();//退出程序
18.                break;
19.            case 1:adsbook.printInfo();//查看联系人
20.                break;
21.            case 2:AddadsMan(adsbook);//添加联系人
22.                break;
23.            case 3:DeleadsMan(adsbook);//删除联系人
24.                break;
25.            case 4:FindadsMan(adsbook);//查找联系人
26.                break;
27.            case 5:ModifyadsMan(adsbook);//修改联系人
28.                break;
29.            case 6:ClearadsMan(adsbook);//清空联系人
30.                break;
31.
32.        }
33.        system("pause");//防止闪退
```



```

34.         system("cls");//清除上一次，保持主界面
35.     }
36.     return 0;
37. }

```

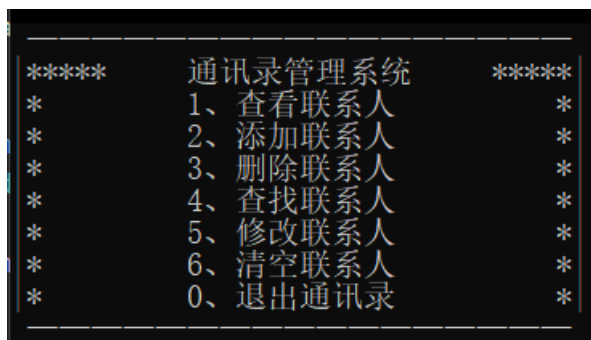


图 3.1.1 功能菜单界面

3.2 系统功能实现

● 退出程序

退出系统操作简单，只需提示用户是否确认，确认则提示用户 3 秒后退出，使用 `Sleep` 函数以及 `exit` 函数，用户选择否即不做任何操作。

代码如下：

```

1. //主函数中调用的退出程序函数
2. void Exitprogram()
3. {
4.     int flag;//记录判断
5.     cout << "是否确认(1、是 2、否): "; cin >> flag;
6.     if (flag == 1)//确认后休眠三秒后退出程序
7.     {
8.         cout << "三秒后将退出程序" << endl;
9.         Sleep(3000);//系统休眠三秒，随后退出
10.        exit(0);//退出程序
11.    }
12. }

```

● 查看联系人

该功能无需用户操作，选择后自动遍历容器中所有成员，有则所有打印，无则提示“当前联系人为空”。

代码如下：

```

1. //打印所有的联系人
2. bool adsBook::printInfo()

```

```

3. {
4.     if (book.size() == 0)
5.     {
6.         cout << "当前联系人为空!" << endl;
7.         return false;
8.     }
9.     for (int i = 0; i < book.size(); i++)
10.    {
11.        cout << "编号: " << i + 1;
12.
13.        book[i].PrintInfo();//调用 Info 类中输出函数
14.    }
15.    return true;
16. }

```

信息类中的打印成员函数：

```

1. //打印信息
2. void Info::PrintInfo()
3. {
4.     cout << "\t 姓名: " << Name << "\t 出生日期: "
5.         << birth.getBirth() << "\t 电话号码: " << Tellnum
6.         << "\t 工作单位: " << Workplace << "\t 电子邮箱: "
7.         << Adsmail << endl;
8. }

```



图 3.2.1 查看联系人功能

● 添加联系人

用户根据提示依次输入联系人信息，系统创建一个临时的 Info 类，调用通讯录类中的添加成员函数添加到通讯录对象中，随后调用类中 inFile 函数写入文件中。

代码如下：

```

1. void AddadsMan(adsBook &adsbook)
2. {
3.     //存下待添加联系人信息

```

```

4.     string name,tell,work,mail;
5.     int y, m, d;
6.
7.     //输入待添加联系人的信息
8.     cout << "请输入姓名: "; cin >> name; cout << endl;
9.     cout << "请输入出生日期（使用空格分隔）: "; cin >> y >> m >> d; cout << endl;
10.    cout << "请输入电话号码: "; cin >> tell; cout << endl;
11.    cout << "请输入工作单位: "; cin >> work; cout << endl;
12.    cout << "请输入电子邮箱: "; cin >> mail; cout << endl;
13.
14.    Info temp(name, y, m, d, tell, work, mail); //创建临时信息类
15.
16.    //调用类中方法添加
17.    adsbook.addInfo(temp);
18. }

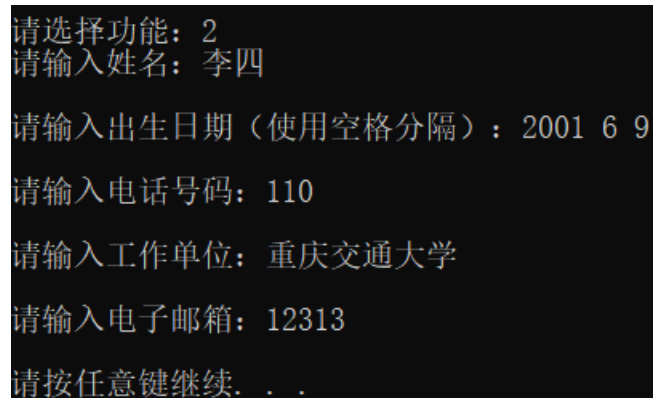
```

通讯录类添加联系人函数：

```

1. //添加一个信息
2. void adsBook::addInfo(const Info & in)
3. {
4.     book.push_back(in);
5.     inFile();
6. }

```



```

请选择功能: 2
请输入姓名: 李四

请输入出生日期（使用空格分隔）: 2001 6 9

请输入电话号码: 110

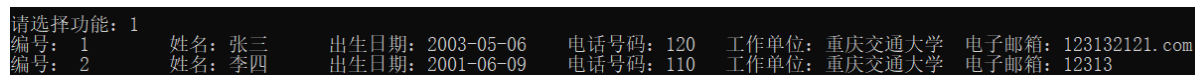
请输入工作单位: 重庆交通大学

请输入电子邮箱: 12313

请按任意键继续. . .

```

图 3.2.2 用户输入信息



```

请选择功能: 1
编号: 1 姓名: 张三 出生日期: 2003-05-06 电话号码: 120 工作单位: 重庆交通大学 电子邮箱: 123132121.com
编号: 2 姓名: 李四 出生日期: 2001-06-09 电话号码: 110 工作单位: 重庆交通大学 电子邮箱: 12313

```

图 3.2.3 信息添加成功

● 删除联系人

系统首先调用一次打印函数打印出所有联系人，无联系人则退出该功能，反之提示用户输入待删除联系人编号，随后调用通讯录类中删除联系人函数，随后写入文件，根据返回值，判断是否删除成功。

代码如下：

```
1. void DeleadsMan(adsBook& adsbook)
2. {
3.     int index;
4.
5.     //打印所有信息再提示用户输入编号进行删除，ok 标记联系人是否为空
6.     bool ok = adsbook.printInfo();
7.     if (ok)
8.     {
9.         cout << "请输入待删除联系人编号: ";
10.        cin >> index;
11.    }
12.    else return;
13.
14.    //ok 读取是否成功
15.    ok = adsbook.deleInfo(index - 1);
16.    if (ok)cout << "删除成功" << endl;
17.    else cout << "删除失败" << endl;
18. }
```

通讯录删除联系人函数：

```
1. bool adsBook::deleInfo(int index)
2. {
3.     //判断选择是否正确
4.     if (index > book.size() || index < 0)return false;
5.
6.     for (int i = 0; i < book.size(); i++)
7.     {
8.         if (i == index)
9.         {
10.            book.erase(book.begin() + i);//删除
11.            inFile();//写入文件
12.            return true;
13.        }
14.    }
15. }
```

```

请选择功能：3
编号： 1      姓名：张三      出生日期：2003-05-06
编号： 2      姓名：李四      出生日期：2001-06-09
请输入待删除联系人编号：2
删除成功
请按任意键继续. . .

```

图 3.2.4 删除联系人

● 查找联系人

用户选择根据姓名或电话号码进行查找，系统判断用户的选择，并且调用通讯录类中查找函数 `findInfo`，用数组存下所有满足添加的联系人，随后进行打印输出。

代码如下：

```

1. //主函数中调用的查找联系人函数
2. void FindadsMan(adsBook& adsbook)
3. {
4.     string n;
5.     int coe;
6.     //输入信息进行删除
7.     cout << "请选择以何种方式查找联系人（0、姓名 1、电话）"; cin >> coe; cout << endl;
8.     cout << "请输入待查找联系人信息："; cin >> n; cout << endl;
9.
10.    //调用类中方法查找
11.    adsbook.findInfo(n, coe);
12. }

```

通讯录类中查找成员函数：

```

1. void adsBook::findInfo(string n, int wayofdele)
2. {
3.
4.     int reind[2000] ; int renum = 0; //记录多条联系人
5.     if (wayofdele == RESARCHWAY::NamePatterm)
6.     {
7.         for (int i = 0; i < book.size(); i++)
8.         {
9.             if (book[i].getName() == n) reind[renum++] = i;
10.
11.         }
12.     }
13.     else if (wayofdele == RESARCHWAY::TellPattern)

```

```

14.     {
15.         for (int i = 0; i < book.size(); i++)
16.         {
17.             if (book[i].getTellnum() == n)reind[renum++] = i;
18.         }
19.     }
20.     if (renum == 0)//未找到时直接返回
21.     {
22.         cout << "未能找到相对应联系人!" << endl;
23.         return ;
24.     }
25.     cout << "共找到 " << renum << " 条联系人记录" << endl;
26.     for (int i = 0; i < renum; i++)
27.     {
28.         book[reind[i]].PrintInfo();
29.     }
30.     return ;
31. }

```

```

请选择功能：4
请选择以何种方式查找联系人（0、姓名 1、电话）0
请输入待查找联系人信息：张三
共找到 1 条联系人记录
      姓名：张三      出生日期：2003-05-06      电话号码：120
请按任意键继续. . .

```

图 3.2.5 查找联系人

● 修改联系人

用户根据提示信息依次输入待修改的信息，根据用户输入编号进行对通讯录类中成员修改，调用信息类中修改数据函数，随后写入文件，根据返回值输出是否修改成功。

代码如下：

```

1. //主函数中调用的修改联系人信息函数
2. void ModifyadsMan(adsBook& adsbook)
3. {
4.     string name, tell, work, mail;
5.     int y, m, d;
6.     int index;
7.
8.     //先打印出所有联系人

```

```
9.     bool ok = adsbook.printInfo();
10.    if (!ok)return;
11.
12.    cout << "请输入待修改联系人编号"; cin >> index; cout << endl;
13.
14.    //输入待添加联系人的信息
15.    cout << "请输入姓名: "; cin >> name; cout << endl;
16.    cout << "请输入出生日期（使用空格分隔）: "; cin >> y >> m >> d; cout << endl;
17.    cout << "请输入电话号码: "; cin >> tell; cout << endl;
18.    cout << "请输入工作单位: "; cin >> work; cout << endl;
19.    cout << "请输入电子邮箱: "; cin >> mail; cout << endl;
20.    Info temp(name, y, m, d, tell, work, mail);
21.
22.    //读取是否修改成功
23.    ok=adsbook.modifyInfo(index-1, temp);
24.
25.    if (ok)cout << "修改成功" << endl;
26.    else cout << "修改失败" << endl;
27. }
```

类中函数:

```
1.  bool adsBook::modifyInfo(int index, Info& in)
2.  {
3.      if (index > book.size() || index < 0)//给出的编号超出范围
4.      {
5.          return false;
6.      }
7.      else
8.      {
9.          book[index] = in;
10.         inFile();
11.         return true;
12.     }
13. }
```

```

请选择功能：5
编号： 1      姓名：张三      出生日期：2003-05-06
请输入待修改联系人编号1

请输入姓名：张四

请输入出生日期（使用空格分隔）：2003 5 6

请输入电话号码：120

请输入工作单位：重庆交通大学

请输入电子邮箱：123456.com

修改成功
请按任意键继续. . .

```

图 3.2.6 修改联系人信息

● 清空联系人

该功能较为简单，用户确认是否删除，随后使用 `vector` 容器的 `clear` 函数即可清空，随后写入文件。

代码如下：

```

1. //主函数中调用的清空联系人函数
2. void ClearadsMan(adsBook& adsbook)
3. {
4.     int flag;
5.     cout << "确认是否清空(1、是 2、否)"; cin >> flag; cout << endl;
6.
7.     if (flag)
8.     {
9.         adsbook.clearInfo();
10.        cout << "清除成功!" << endl;
11.    }
12.
13.    else cout << "已取消清空!" << endl;
14. }

```

类中成员函数：

```

1. void adsBook::clearInfo()
2. {
3.     book.clear();
4.     inFile();
5. }

```



```

请选择功能：6
确认是否清空(1、是 2、否)1

清除成功！
请按任意键继续. . .

```

图 3.2.7 清空联系人

```

***** 通讯录管理系统 *****
*      1、查看联系人      *
*      2、添加联系人      *
*      3、删除联系人      *
*      4、查找联系人      *
*      5、修改联系人      *
*      6、清空联系人      *
*      0、退出通讯录      *
*****

请选择功能：1
当前联系人为空！
请按任意键继续. . .

```

图 3.2.8 清空成功

● 辅助或其他功能

首先创建通讯录管理对象时调用的构造函数，作用在读取文件中联系人代码如下：

```

1. adsBook::adsBook()
2. {
3.
4.     fstream In;
5.     In.open("adsmans.txt", ios::in);
6.     string n; int y; int m; int d; string tell; string work; string email;
7.     while (!In.eof())
8.     {
9.         In >> n >> y >> m >> d >> tell >> work >> email;
10.        book.push_back(Info(n, y, m, d, tell, work, email));
11.    }
12.    book.erase(book.end()-1); //最后一行为空，也会被读入
13. }

```

写入文件的函数用于每次对联系人的修改，由于使用了 `exit` 函数，程序不能完整的完成 `main` 函数中的代码，也即对象无法调用析构函数，故不使用析构函数进行文件写入，而采取普通成员函数 `inFile`。

代码如下：

```

1. void adsBook::inFile()
2. {
3.     ofstream out;
4.     out.open("adsmans.txt",ios::trunc|ios::out);
5.     for (int i = 0; i < book.size(); i++)
6.     {
7.         out << book[i].getName() << " " << book[i].getBirthYear() <<
            " " << book[i].getBirthMonth() << " "
8.         << book[i].getBirthDay() << " " << book[i].getTelnum() <
            < " " << book[i].getWorkplace() << " "
9.         << book[i].getAdsmail() << endl;
10.    }
11. }

```

最后，由于其余的成员函数如赋值函数 `setxxx`、读取类中数据函数 `getxxx` 等比较多并且比较简单在此处就不再写出。

4. 测试及测试结果

功能测试

项目/软件	通讯录管理系统			程序版本	V1.0	
功能名称	查看联系人					
测试目的	是否正常打印、是否正确提示无联系人情况、日期是否正常打印					
预置条件	若干联系人情况、无联系人情况					
用例编号	目的	操作步骤	输入数据	期望结果	执行结果	缺陷报告
01	是否正常打印	无	无	按表格样式打印	成功	当名字过长表格不整齐
02	是否提示无联系人	无	无	提示无联系人	成功	无
03	日期是否正常打印	无	11111 1 1	输出类似 yyyyy-mm-dd	失败	当出生年不为 4 位数时 前面会补零或者只能输出 4 位数

项目/软件	通讯录管理系统			程序版本	V1.0	
功能名称	添加联系人					
测试目的	是否添加到文件、添加的数据是否正确					
预置条件	无					
用例编号	目的	操作步骤	输入数据	期望结果	执行结果	缺陷报告
01	是否添加到文件	打开系统进行添加，关闭系统，打开文件查看数据	联系人信息	文件中数据添加	成功	无
02	添加的数据是否正确	系统添加数据，再次打开查看数据	联系人信息	数据完全正确	成功	无
03	无输入是否数据正确	系统中间隔不输入数据	部分联系人信息	不能无数据输入	成功	无

项目/软件	通讯录管理系统			程序版本	V1.0	
功能名称	删除联系人					
测试目的	是否从文件中删除、是否改变其他联系人					
预置条件	若干联系人					
用例编号	目的	操作步骤	输入数据	期望结果	执行结果	缺陷报告
01	是否从文件中删除	打开系统进行删除， 关闭系统，	联系人姓名或电话	文件中已删除联系人	成功	无

		打开文件 查看数据				
02	是否改变其他联系人	系统删除联系人，再次打开查看联系人	联系人信息	其他联系人未被改变	成功	无

项目/软件	通讯录管理系统			程序版本	V1.0	
功能名称	查找联系人					
测试目的	是否正确查找、是否完全查找、是否修改数据					
预置条件	若干联系人、无联系人					
用例编号	目的	操作步骤	输入数据	期望结果	执行结果	缺陷报告
01	是否正确查找	无	无	查找结果与数据文本文件中一致	成功	查找无法有其他操作
02	是否完全查找	无	无	符合条件的的所有联系人均被找出	成功	无
03	是否修改数据	打开数据文本文件	无	数据文本文件没有改变、查找结果不变	成功	无

项目/软件	通讯录管理系统			程序版本	V1.0	
功能名称	修改联系人					
测试目的	是否正确修改、无联系人能否修改、其他联系人是否被修改					
预置条件	若干联系人、无联系人					
用例编号	目的	操作步骤	输入数据	期望结果	执行结果	缺陷报告
01	是否正确修改	修改	另一联系人信息	数据与输入数据一致	成功	无
02	无联系人能否修改	无	无	提示无联系人后，退出该功能	成功	无
03	其他联系人是否被修改	修改	另一联系人信息	只有被修改的联系人数据改变	成功	无

项目/软件	通讯录管理系统			程序版本	V1.0	
功能名称	清空联系人					
测试目的	是否正常清空并写入文件					
预置条件	若干联系人					
用例编号	目的	操作步骤	输入数据	期望结果	执行结果	缺陷报告
01	是否正常清 空并写入文 件	无	无	系统中无联 系人、文件中 无联系人	成功	无

集成测试

项目/软件	通讯录管理系统			程序版本		V1.0	
功能名称	通讯录系统						
测试目的	测试各个功能接口是否完善						
预置条件	若干联系人、无联系人						
用例编号	目的	操作步骤	输入数据	期望结果	执行结果	缺陷报告	
01	测试各个功能接口是否完善	各个功能依次使用	联系人信息	各个功能完善,数据之间流通性好	成功	无	