

Hong Kong Diploma of Secondary Education Examination 2016

Information and Communication Technology

School-based Assessment

Option D: Software Development

Title: Phone Book Directory

School: Cheung Sha Wan Catholic Secondary School

Content

Chapter1 – Introduction

- 1.1 Background
- 2.2 Situation

Chapter2 - Design and Implementation

- 2.1 Brief description
- 2.2 Design of the main program
- 2.3 Design of the sub program
- 2.4 Data structure
- 2.5 User Guide
- 2.6 Problem on producing the program

Chapter3 - Testing

Chapter4 - Conclusion

- 4.1 evaluation
 - 4.2 future plans
 - 4.3 Schedule
 - 4.4 Appendix
- Source Codes

Introduction

1.1 Background

In the 20th Century, with the aid of technology, lives have become more convenient when compared with before. People start to rely on these technology and they don't really need to try to remember many things.

Phone book is one of the examples which could be found in computer or in a mobile phone. People could simply save friends' or important people's contact information. All data could be searched and use at any time if needed. It is free of charge unless you don't have a computer or a mobile phone.

2.2 Situation

In recent days, some people has stated that the present phone book is too old in its version and new functions should be added. As a member of the technical team, I have been assigned to make a newer version of the current phone book. the following will be a report about the phone book I have invented.

Design and Implementation

2.1 Brief description

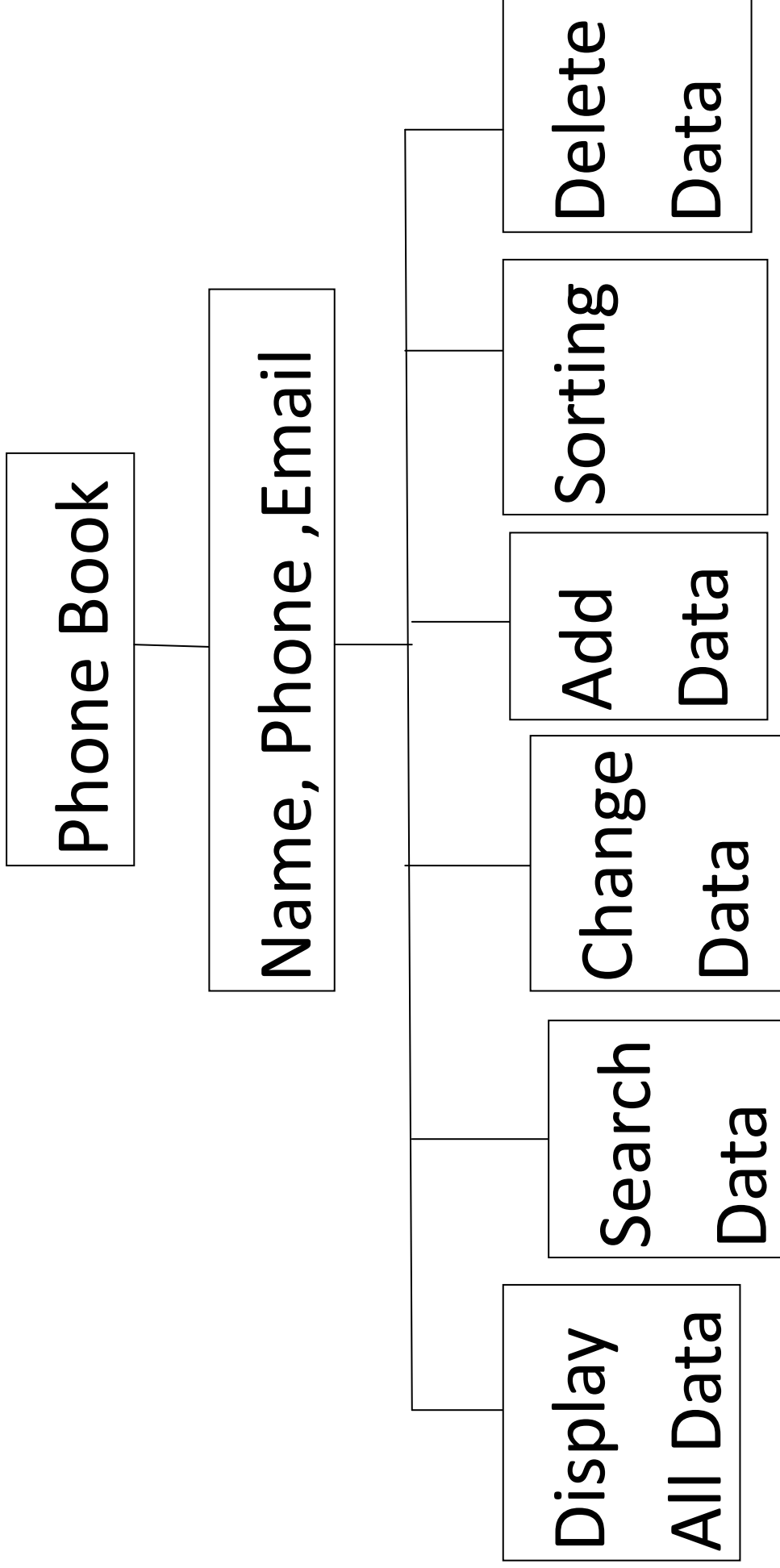
- **Basic requirement of system**

For the system, the required process to overcome the situation is that to make a new phone book program to let people use. There are two main points in which to make sure to be done in order to make this phone book. Firstly, it must be user-friendly. The point user-friendly is the essential condition to attract people continuing to use it. Second, it must be simplified and codes should be changed easily. Every people may have different ideas on their phone book. Program should be edited for satisfactory of customers. So, simplified codes is very important.

- **Characteristics of the program**

The program would be built by command line interface using Pascal. Also, the program is made up of different small-divided procedures. It is to ensure the reusability of the procedures and minimize the number of logical statements and algorithms that needed to be coded. Last but not least, as I have said before, the program should have simplified coding so as to make sure the program could be make changes easily. The program is designed for future use and so changes could be made efficiently by adding extra procedures or modifying little arrangements in the program.

2.2 Design of the main program



2.3 Design of the sub programs

This program is a phone book. It consists of 3 elements, a person's name, her phone number and his email address. There are 6 procedures in the main program.

1. Display all data

It is used to show all the elements which have stored in the text file. All data will be showed in the program.

2. Search Data

It is used to search a person's data which have stored in the text file. A new feature is that if the name searched is not found in the text file, you will be asked if you want to add data. If the answer is yes, it will run the procedure of add data.

3. Change data

It is used to change a person's data which have stored in the text file. When you choose this function, a sub title page will be showed. There will be 3 choices to choose, change name, change phone number and change email address. You could choose any data you want to change in your phone book. It is to make the procedure become more convenient to use.

4. Add data

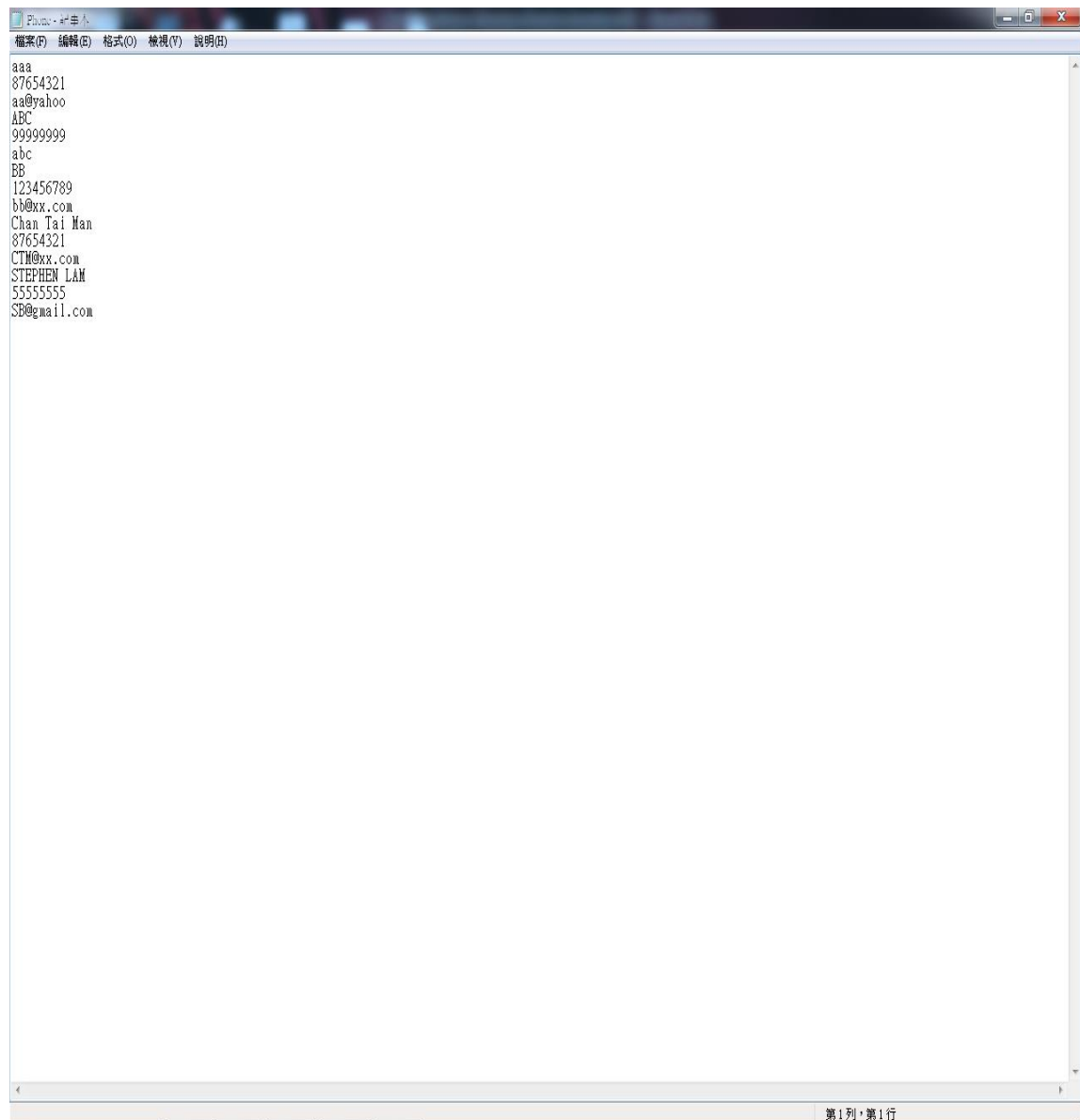
It is used to add a person's data to store in the text file. A procedure write file is used in the main program. You could simply write the person's data you want to added. The data will automatically saved to the text file which stores the phone book data.

5. Sorting

It is used to change the position of the data by alphabetical order. The data are sorted by name after this procedure runs. It is to make the program more user friendly.

6. Delete data

It is used to delete current data stored on the text file. Users only needs to type the person's name whose data has to be delete. All of this person will be deleted automatically.



This is the text file which stores all data for the phone book. For each person, the first column is his/her name, the second column is the phone number and the third column is the email address respectively.

2.4 Data Structure

```
const   Max = 100;
```

```
        datafile = 'Phone.txt';
```

```
var   Name : array[1..Max] of string;
```

```
        Phone : array[1..Max] of string;
```

```
        Email : array[1..Max] of string;
```

```
        N, Choice : integer;
```

```
        {Main Program}
```

```
        NewName : integer;
```

```
        NewPhone : integer;
```

```
        NewEmail : integer;
```

```
        Infile : text;
```

```
        {Read File}
```

```
        Outfile : text;
```

```
        {Write File}
```

```
        temp_name, temp_phone, temp_email : string;
```

```
        swap : boolean;
```

```
        {Sorting}
```


FindName : string;

{Change Data, Delete Data}

Locate : integer;

{Delete Data}

2.5 User Guide

Firstly, users should create a new text file named 'Phone'.

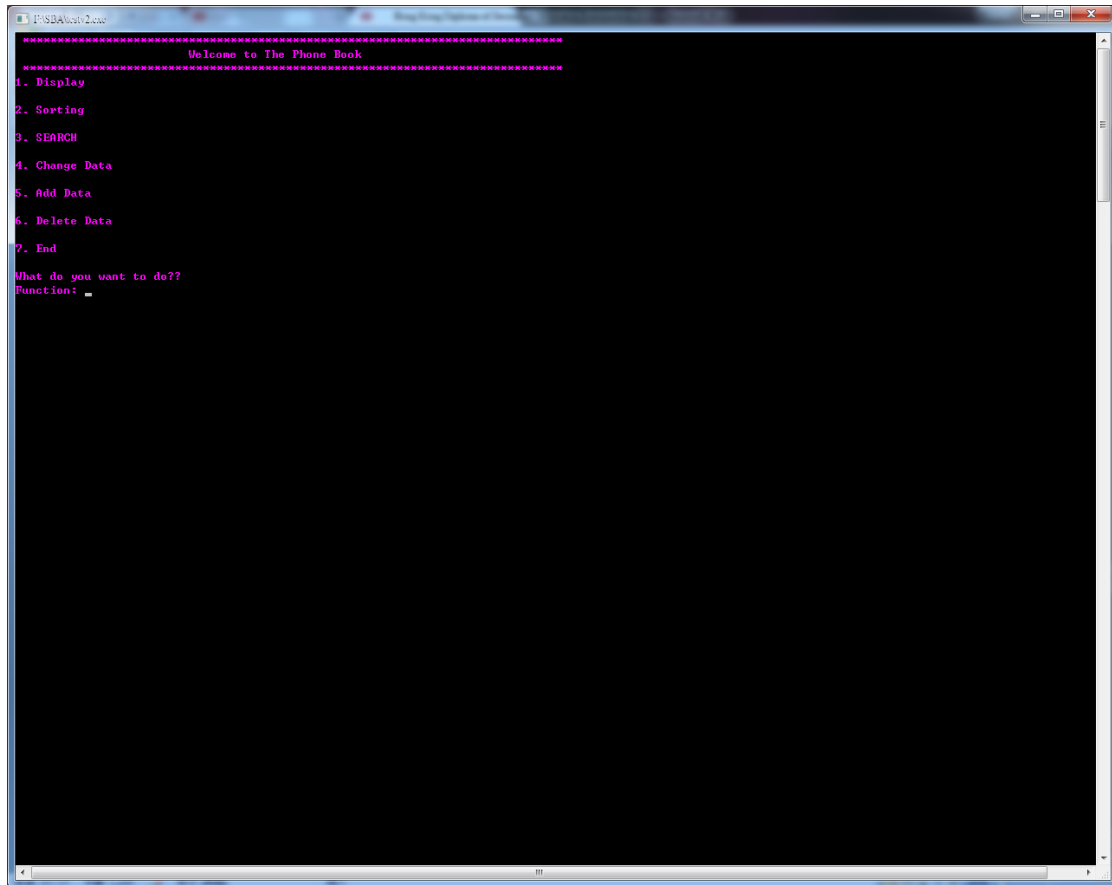
Otherwise, this program may not run successfully.

Then, users may refer to **2.3 Design of the sub programs.**

It all tells the functions of each procedures.

2.6 Problem on producing the program

In here, I will show some of the bugs or problems and the final product of the program.

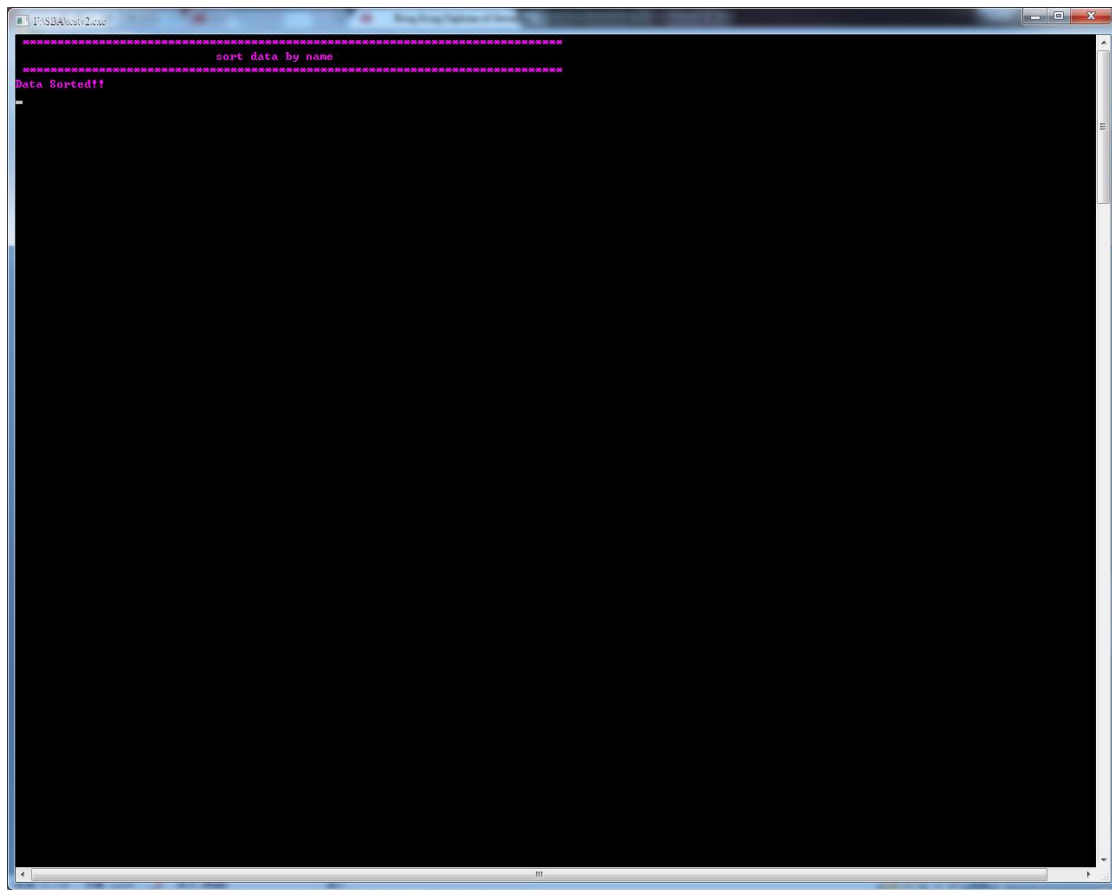


```
PSBAdev2.exe
=====
Welcome to The Phone Book
=====
1. Display
2. Sorting
3. SEARCH
4. Change Data
5. Add Data
6. Delete Data
7. End
What do you want to do??
Function: _
```

This is the title page.

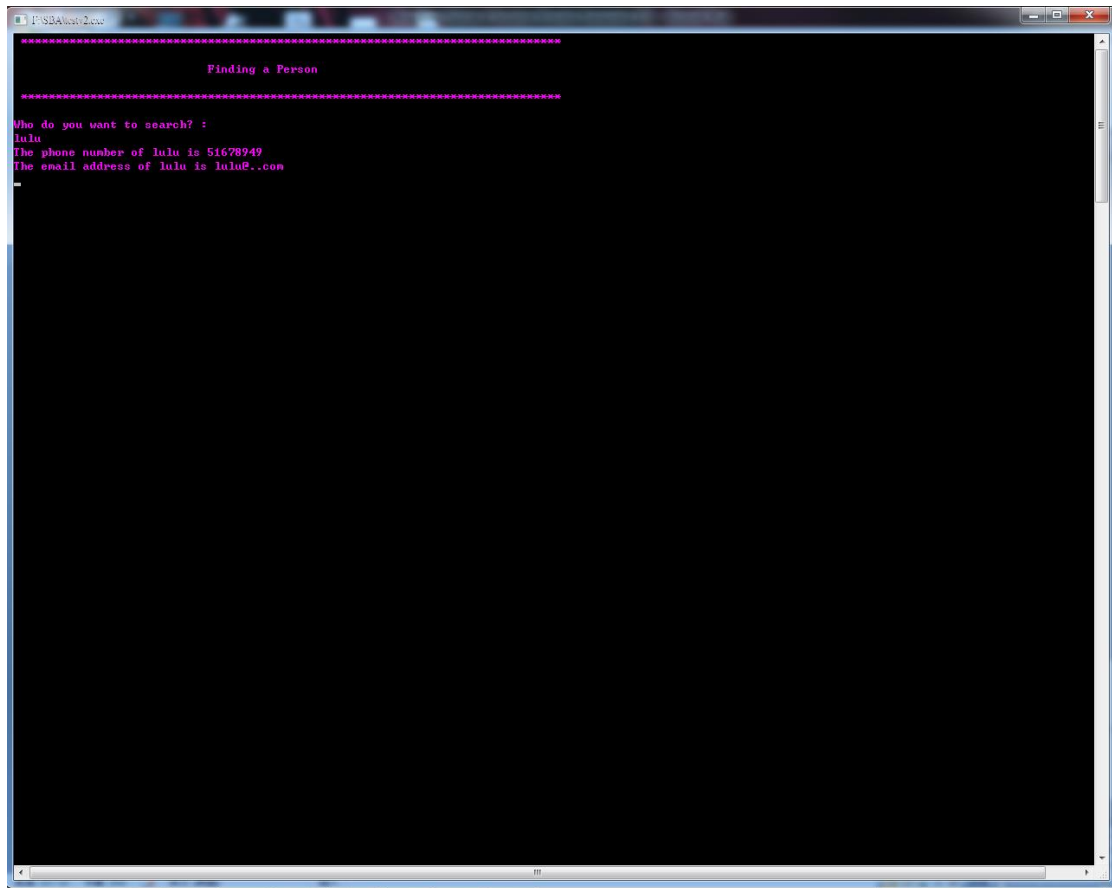
```
=====
                        All People You Can Find Here
=====
Data in Your Phone Book:
The phone no. of aaa is 87654321
The email address of aaa is aa@yahoo
The phone no. of ABC is 99999999
The email address of ABC is abc
The phone no. of BB is 123456789
The email address of BB is bb@xx.com
The phone no. of Chan Tai Man is 87654321
The email address of Chan Tai Man is CTM@xx.com
The phone no. of LULU is 51678949
The email address of LULU is luluf.com
The phone no. of STEPHEN LAM is 55555555
The email address of STEPHEN LAM is SB@gmail.com
-
```

This is the page to display all the data stored on the 'Phone.txt'.



```
***** sort data by name *****
Data Sorted!
```

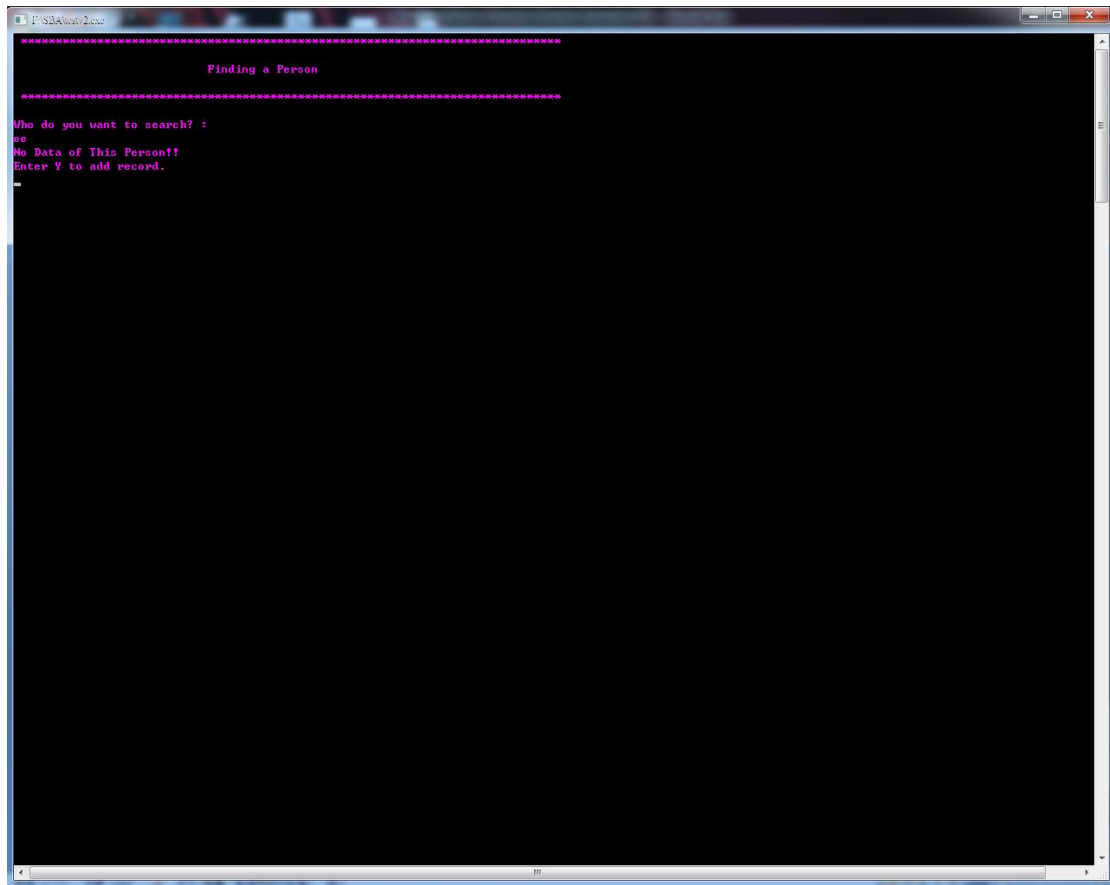
This is the page where sorting performs. When the sorting procedure runs successfully, the sentence 'Data Sorted!' will appear.



```
=====
                          Finding a Person
=====

Who do you want to search? :
lulu
The phone number of lulu is 51678949
The email address of lulu is lulu@.com
```

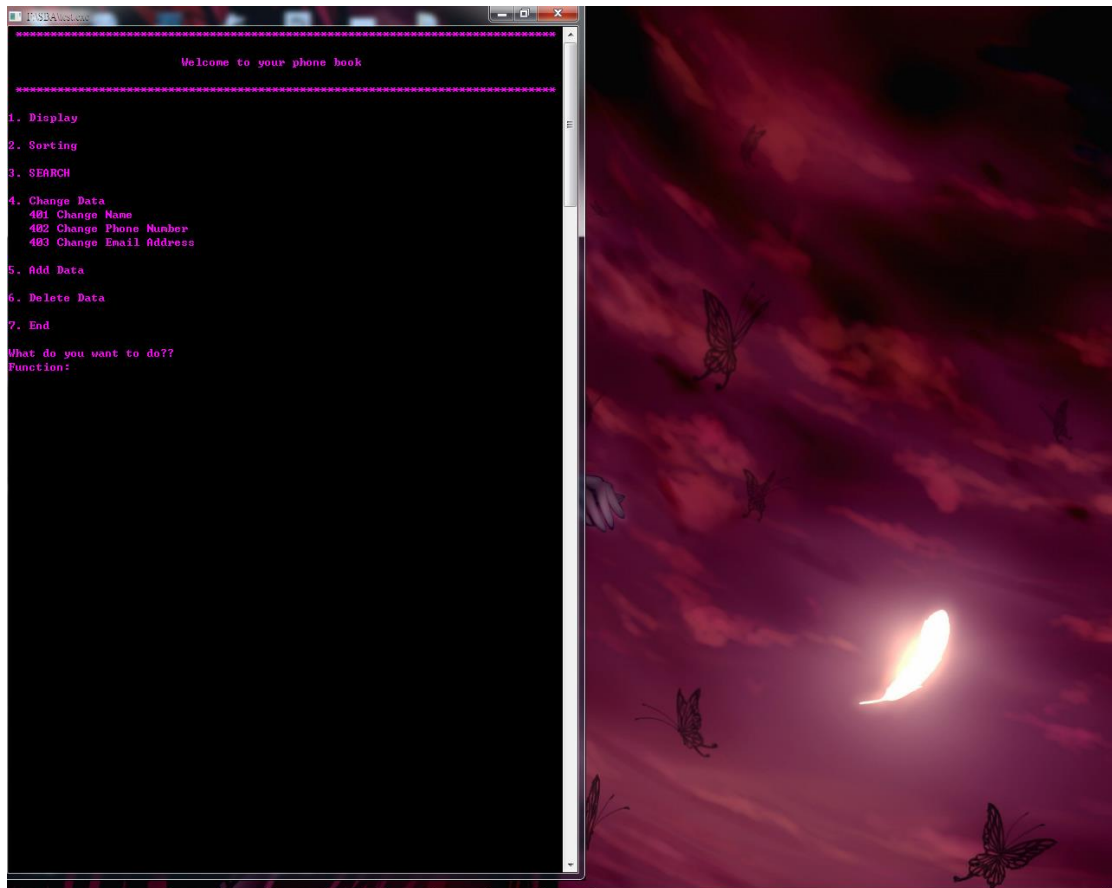
This is the procedure of searching. At the very beginning when producing this procedure, I could not search the right person's data when the name typed, In fact, the very last person's data on the 'Phone.txt' is shown in the program. Of course, the problem has been solved later on.



```
=====
                          Finding a Person
=====
Who do you want to search? :
as
No Data of This Person!!
Enter Y to add record.

```

This is also the procedure of searching. As I have said before, when the name searched is not on the list of the phone book data, a question will be asked that whether users want to add data. At first, I used the selection statement (If .. Then). However I found that it is different from the expected output. It cannot goto the procedure of add data after the user say yes to the question. The reason is that the program cannot detect the procedure add data as I putted it under the procedure of search data. The problem has been solved later.

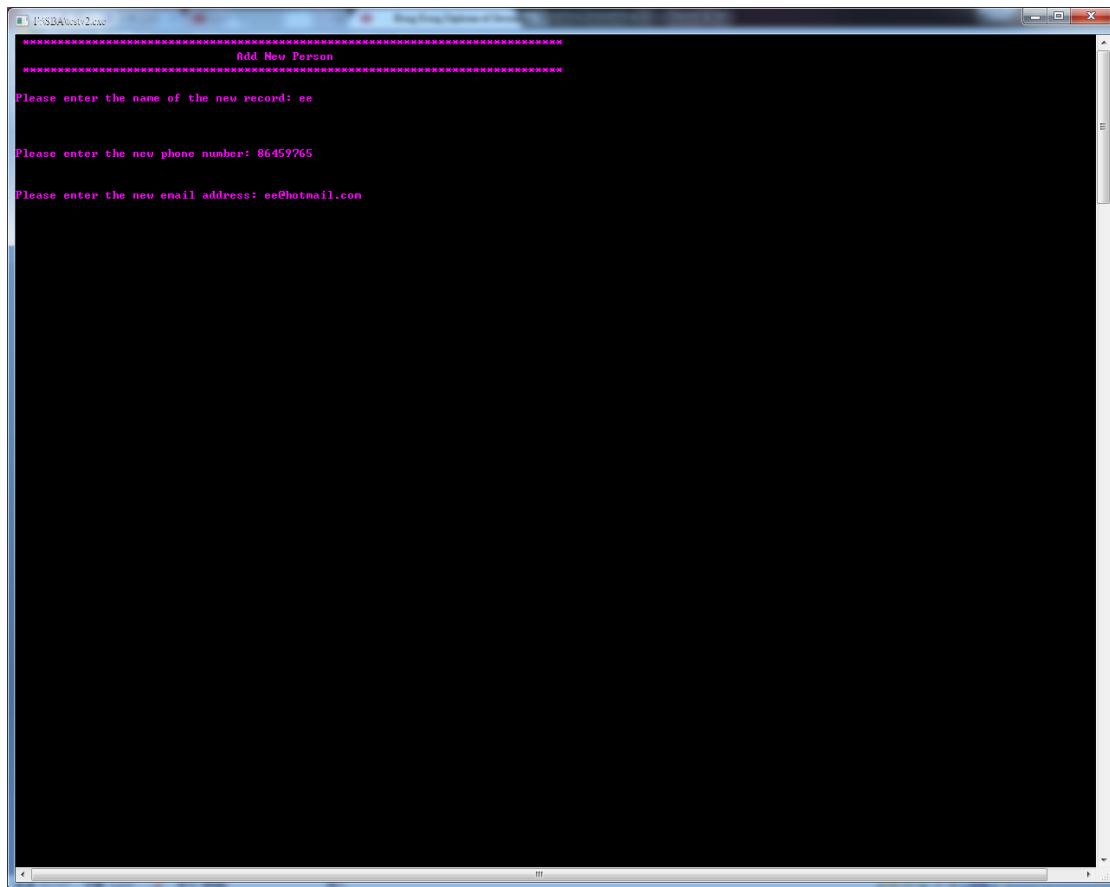


This is the early title page of the phone book. As we can see that the procedure change data has consists of three elements. The case of statement is used and named 401, 402, 403. However It was found that the way is not user-friendly. So, here are the latest version:

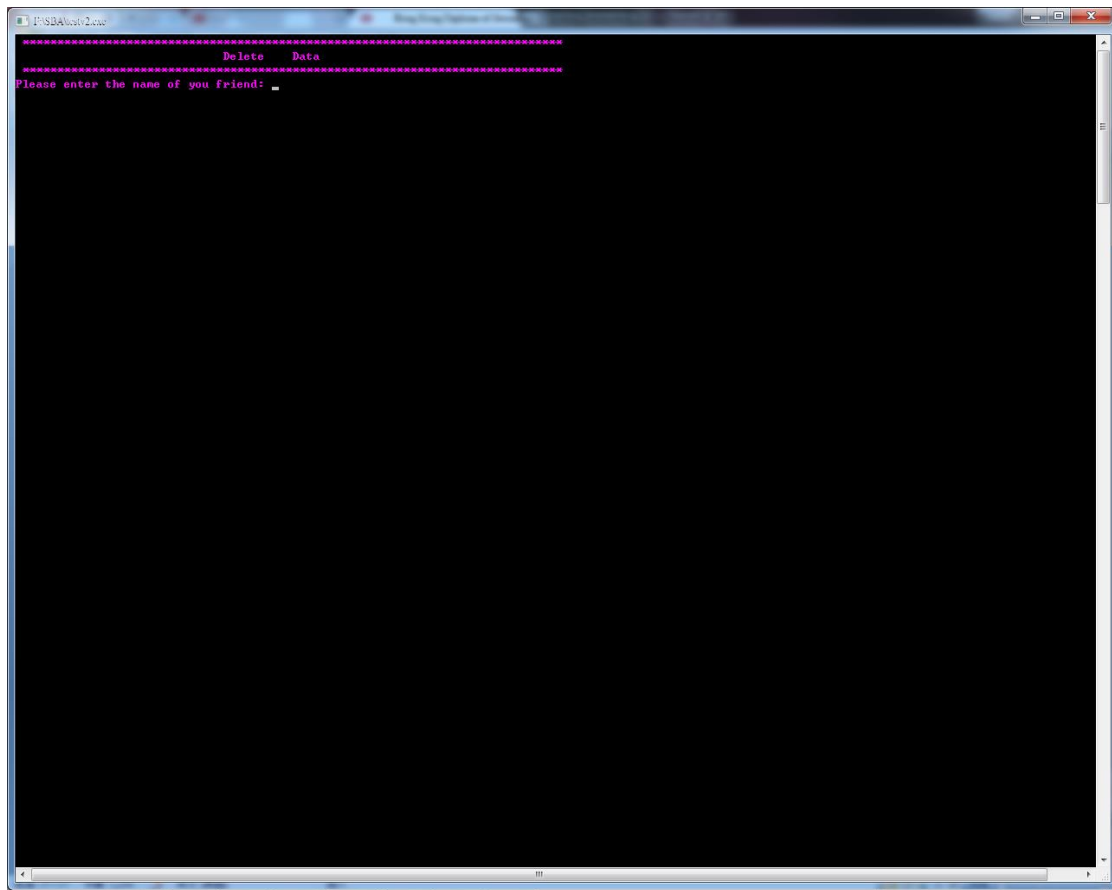
```
PSBAdmin2.exe
=====
Welcome to The Phone Book
=====
1. Display
2. Sorting
3. SEARCH
4. Change Data
5. Add Data
6. Delete Data
7. End
What do you want to do??
Function:
```

```
PSBAdmin2.exe
=====
Change Data
=====
1. Name Change
2. Phone Number Change
3. Email Address Change
4. End
Function:
```


In the above, when 4(Change data) is typed, It will go to a sub title page which user could choose which type of data they want to make amendments. It seems to have a better looking than before.



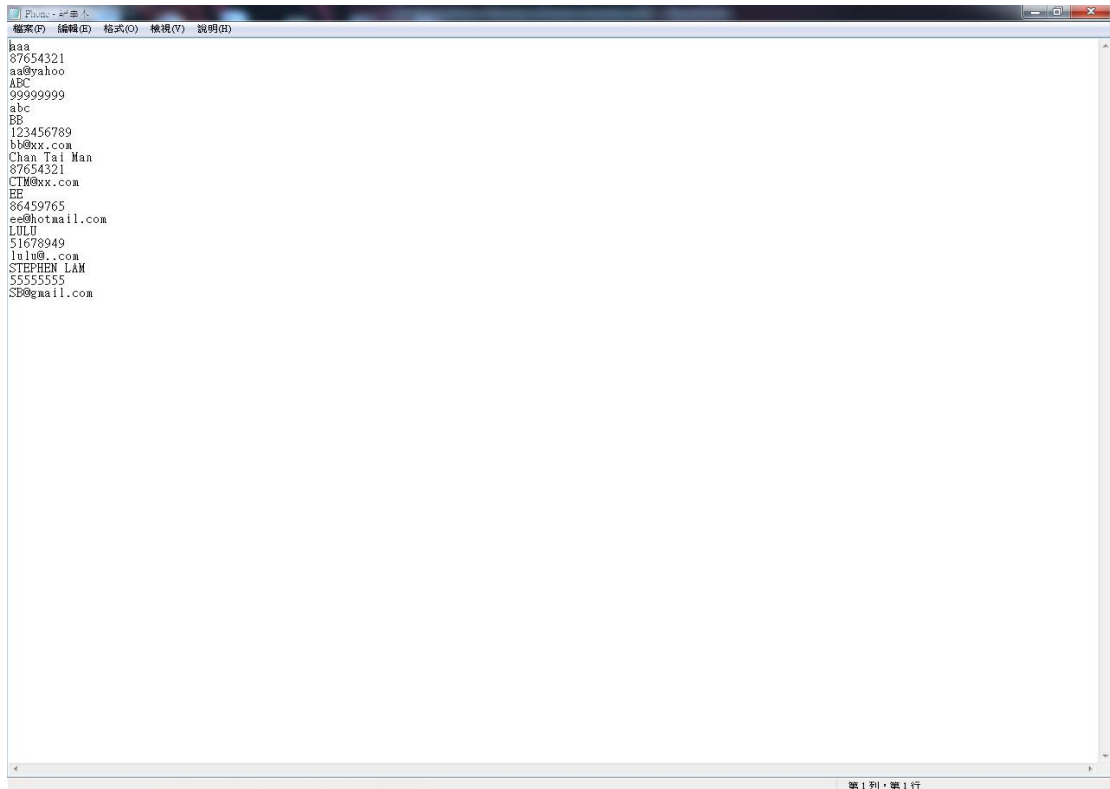
This is the procedure of add data. Users are required to type the name, phone number and email address for the new data. There are some problems of digit validation that people can type more than 8 numbers for the phone number and non letters can be typed for names. I have added a selection statement (If .. Then) to make sure the data typed is valid.



This is the procedure of delete data. Users are required to type the name whose are going to delete. The person's data will be deleted automatically.

Testing

The phone book program is set to open a text file to store all the phone data. So, I opened a text file "Phone.txt" for the testing below.

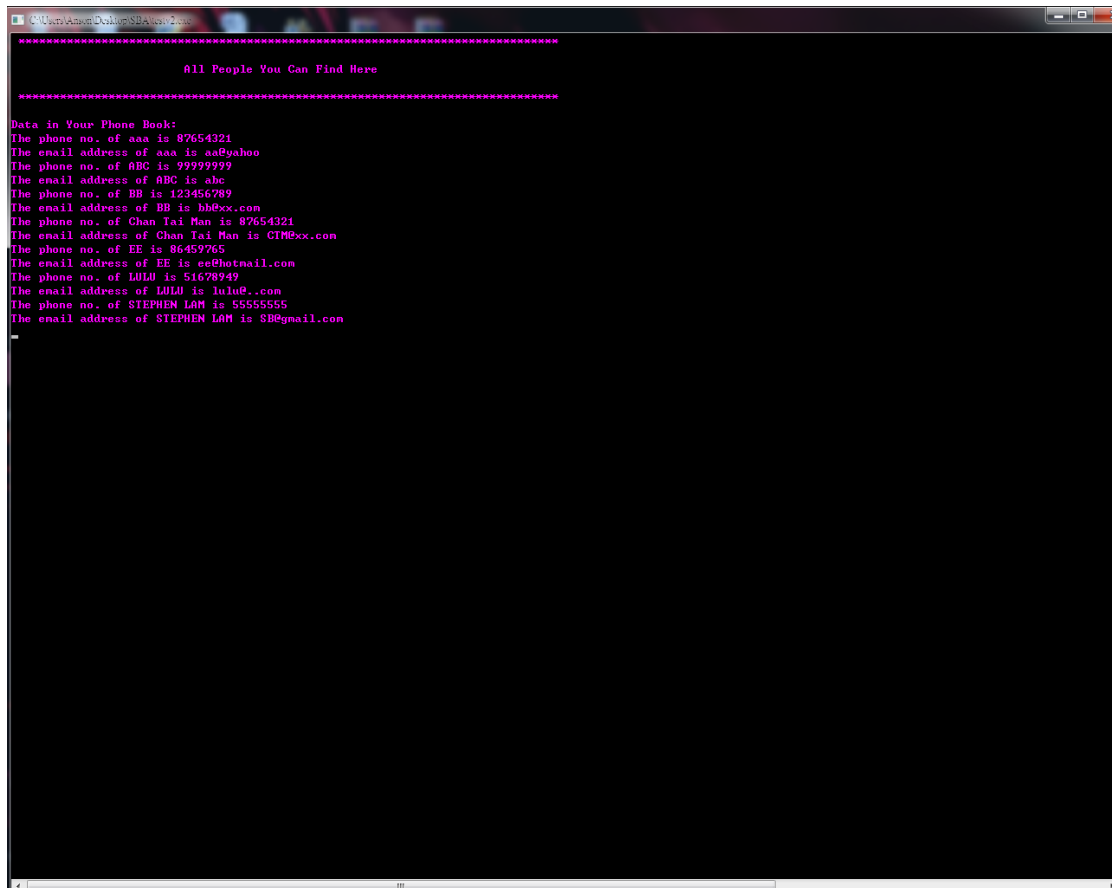


A screenshot of a Notepad window titled "Phone.txt - 47 字元". The window contains a list of names and phone numbers. The text is as follows:

```
aaa  
87654321  
aa@yahoo  
ABC  
99999999  
abc  
BB  
123456789  
bb@xx.com  
Chan Lai Man  
87654321  
CTW@xx.com  
BB  
86459765  
ee@hotmail.com  
LULU  
51678949  
lulu@.com  
STEPHEN LAM  
55555555  
SB@gmail.com
```

The status bar at the bottom right indicates "第 1 列, 第 1 行" (Line 1, Column 1).

Test 1 : Display data

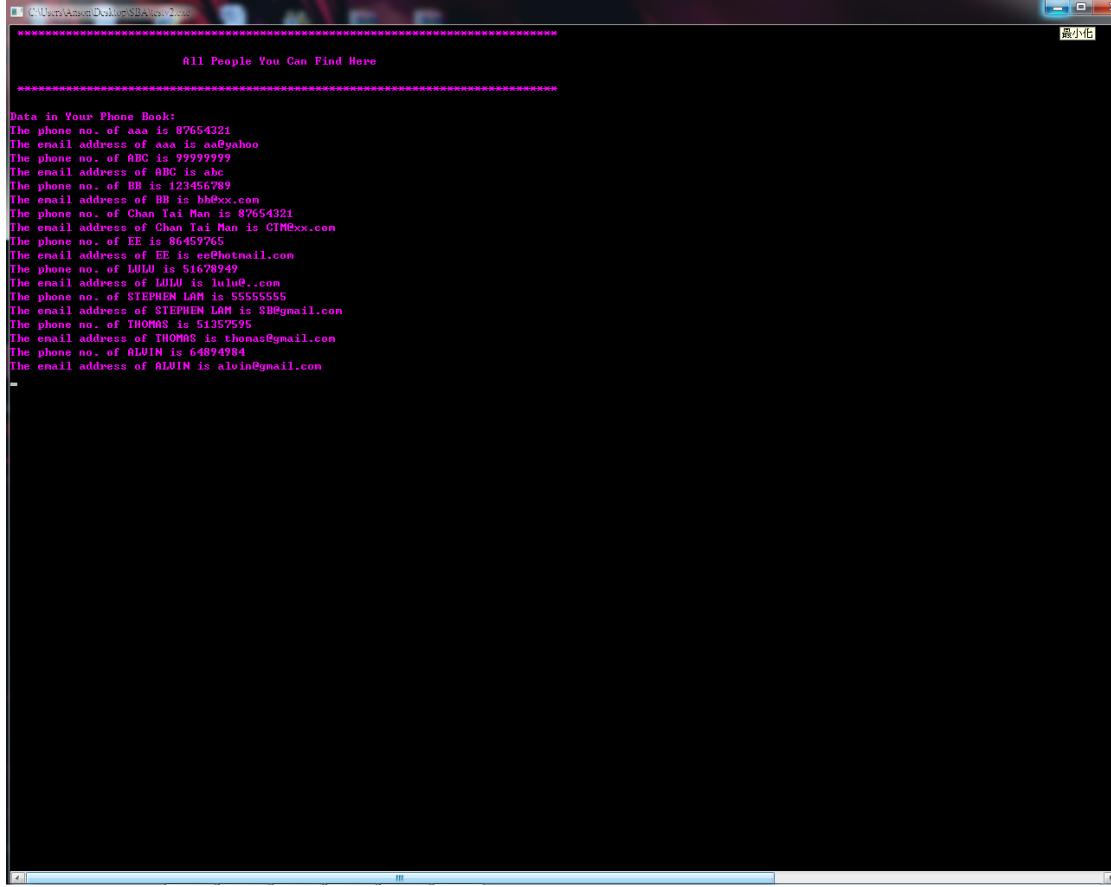


```
*****
All People You Can Find Here
*****

Data in Your Phone Book:
The phone no. of aaa is 87654321
The email address of aaa is aa@yahoo
The phone no. of ABC is 99999999
The email address of ABC is abc
The phone no. of BB is 123456789
The email address of BB is bb@xx.com
The phone no. of Chan Tai Man is 87654321
The email address of Chan Tai Man is CTH@xx.com
The phone no. of EE is 86459765
The email address of EE is ee@hotmail.com
The phone no. of LULU is 51678949
The email address of LULU is lulu@.com
The phone no. of STEPHEN LAM is 55555555
The email address of STEPHEN LAM is SB@gmail.com
```

All the data stored in the "Phone.txt" are shown. When the display data function is used, the data are shown correctly and successfully with respect to the text file.

Test 2 : Sorting



```
=====
All People You Can Find Here
=====

Data in Your Phone Book:
The phone no. of aaa is 87654321
The email address of aaa is aa@yahoo
The phone no. of ABC is 99999999
The email address of ABC is abc
The phone no. of BB is 123456789
The email address of BB is bb@xx.com
The phone no. of Chan Tai Man is 87654321
The email address of Chan Tai Man is CIME@xx.com
The phone no. of EE is 86459765
The email address of EE is ee@hotmail.com
The phone no. of LULU is 51678949
The email address of LULU is lul@.com
The phone no. of STEPHEN LAM is 55555555
The email address of STEPHEN LAM is SB@gmail.com
The phone no. of THOMAS is 51357595
The email address of THOMAS is thomas@gmail.com
The phone no. of ALVIN is 64894984
The email address of ALVIN is alvin@gmail.com
-
```

In the phone book, I used the add data function to add a person named Alvin and his data. However, it is not placed in a correct alphabetical order.

```
sort data by name

Data Sorted!!
```

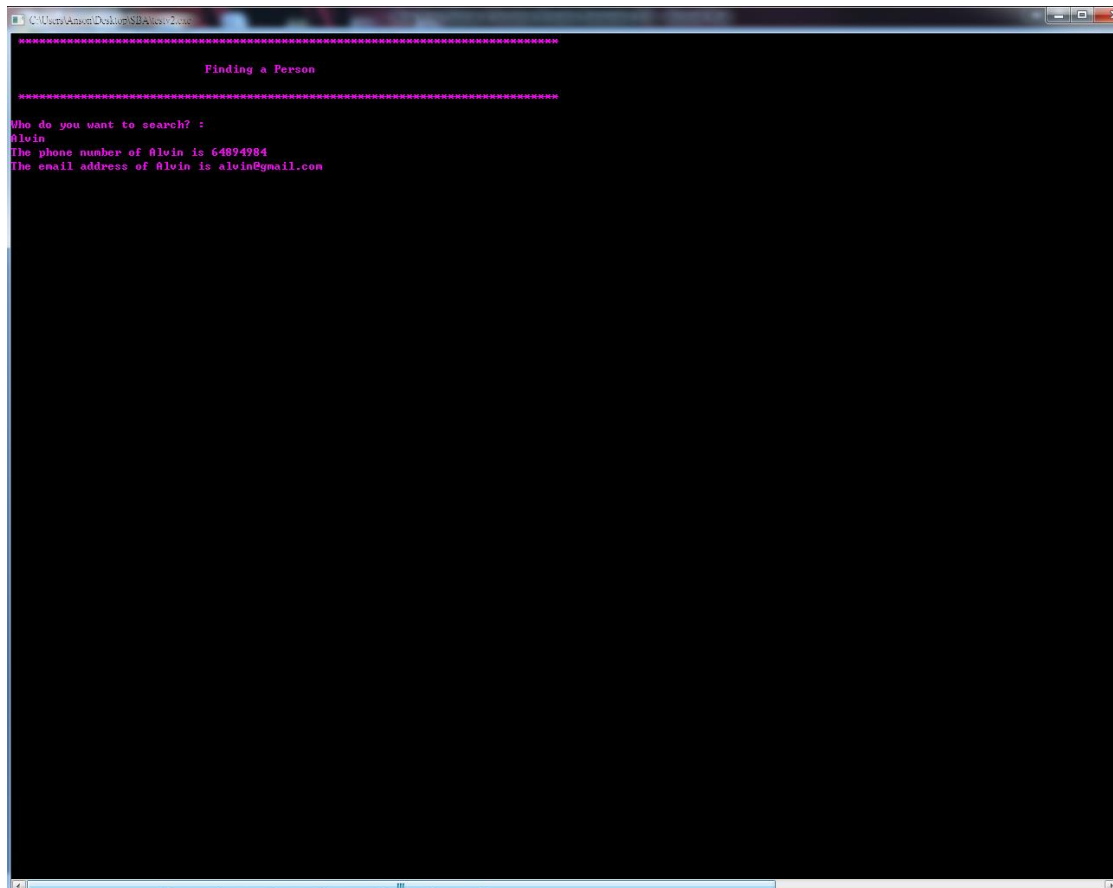
It is the message shown after processing the sorting function.

```
All People You Can Find Here

Data in Your Phone Book:
The phone no. of aaa is 87654321
The email address of aaa is aa@yahoo
The phone no. of abc is 999999999
The email address of abc is abc
The phone no. of ALVIN is 54894984
The email address of ALVIN is alvin@gmail.com
The phone no. of BB is 123456789
The email address of BB is bb@xx.com
The phone no. of Chan Tai Man is 87654321
The email address of Chan Tai Man is CTH@xx.com
The phone no. of EE is 86459765
The email address of EE is ee@hotmail.com
The phone no. of LUIJ is 51598949
The email address of LUIJ is luij@.com
The phone no. of STEPHEN LAM is 55555555
The email address of STEPHEN LAM is SB@gmail.com
The phone no. of THOMAS is 51352595
The email address of THOMAS is thomas@gmail.com
```

The data will be successfully ordered to alphabetical order by name.

Test 3 : Search



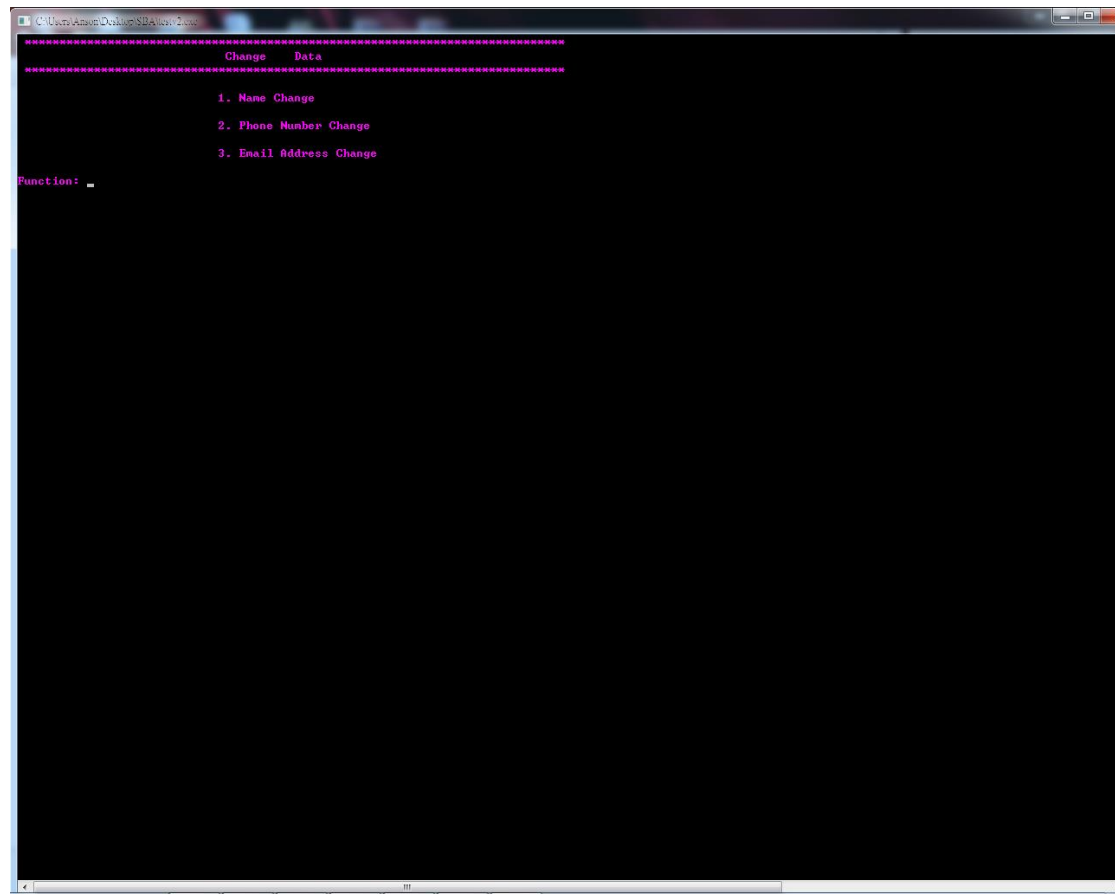
```

C:\Users\Amr\Desktop\SD\Amr\1\Amr1.exe
=====
                          Finding a Person
=====
Who do you want to search? :
Alvin
The phone number of Alvin is 64894984
The email address of Alvin is alvin@gmail.com

```

The data of Alvin is added just now. After entering Alvin's name, his data will be shown successfully.

Test 4 : Change Data



After pressing the change data function, another page will appear and there are 3 options to choose.


```
C:\Users\Alvin\Desktop>python3.py
=====
Making Amendments On Name
=====
Please enter a name :
ALVIN
The name of ALVIN is ALVIN
Enter the new name : ALCIN
```

I change the name of Alvin to Alcin.

```
C:\Users\Alvin\Desktop>python3.py
=====
All People You Can Find Here
=====
Data in Your Phone Book:
The phone no. of aaa is 87654321
The email address of aaa is aa@yahoo
The phone no. of abc is 99999999
The email address of abc is abc
The phone no. of ALCIN is 64894984
The email address of ALCIN is alvin@gmail.com
The phone no. of BB is 123456789
The email address of BB is bb@xx.com
The phone no. of Chan Tai Man is 87654321
The email address of Chan Tai Man is CIME@xx.com
The phone no. of EE is 86459765
The email address of EE is ee@hotmail.com
The phone no. of LULU is 51678949
The email address of LULU is lulu@.com
The phone no. of STEPHEN LAM is 55555555
The email address of STEPHEN LAM is SB@gmail.com
```

The name of Alvin has been changed successfully.

```
C:\Users\Amin\Desktop>Python 2.7.11
Python 2.7.11 Shell
=====
Peking Residents On Phone Number
=====
Please enter a name :
ALCIN
The phone number of ALCIN is 64894984
Enter the new phone number : 64768949
```

For changing the phone number, I changed Alcin's phone number to 64768949.

```
C:\Users\Amin\Desktop>Python 2.7.11
Python 2.7.11 Shell
=====
Data in Your Phone Book:
=====
The phone no. of aas is 87654321
The email address of aas is aas@python
The phone no. of oac is 99999999
The email address of oac is oac
The phone no. of ALCIN is 64768949
The email address of ALCIN is alcin@gmail.com
The phone no. of BB is 123456789
The email address of BB is bb@ccc.com
The phone no. of Chan Tai Pan is 87654321
The email address of Chan Tai Pan is CTP@ccc.com
The phone no. of EE is 86459765
The email address of EE is ee@bbsmail.com
The phone no. of LULU is 61678949
The email address of LULU is lulul@.com
The phone no. of STEPHEN LOP is 65155555
The email address of STEPHEN LOP is SLP@qmail.com
```

The phone number has changed successfully.

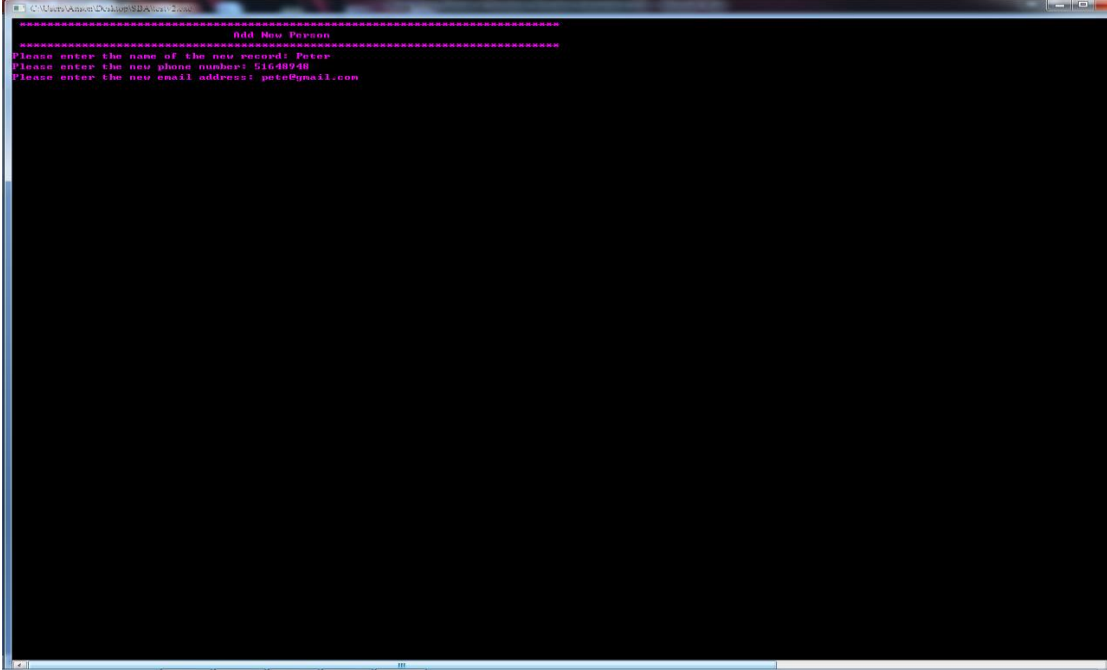
```
C:\Users\Kevin\Desktop>python3.py
=====
Making Amendments On Email Address
=====
Please enter a name :
ALCIN
The email address of ALCIN is alvin@gmail.com
Enter the new email address : alcin@gmail.com
```

For changing email address, I changed Alcin's email address to alcin@gmail.com.

```
C:\Users\Kevin\Desktop>python3.py
=====
All People You Can Find Here
=====
Data in Your Phone Book:
The phone no. of aaa is 87654321
The email address of aaa is aa@yahoo
The phone no. of abc is 99999999
The email address of abc is abc
The phone no. of ALCIN is 64768949
The email address of ALCIN is alcin@gmail.com
The phone no. of BB is 123456789
The email address of BB is bb@ccc.com
The phone no. of Chan Tai Man is 87654321
The email address of Chan Tai Man is CTM@ccc.com
The phone no. of EE is 86459765
The email address of EE is ee@hotmail.com
The phone no. of LLLL is 51678949
The email address of LLLL is lcll@.com
The phone no. of STEPHEN LAM is 55555555
The email address of STEPHEN LAM is sbb@gmail.com
```

The email address has been changed successfully.

Test 5 : Add data

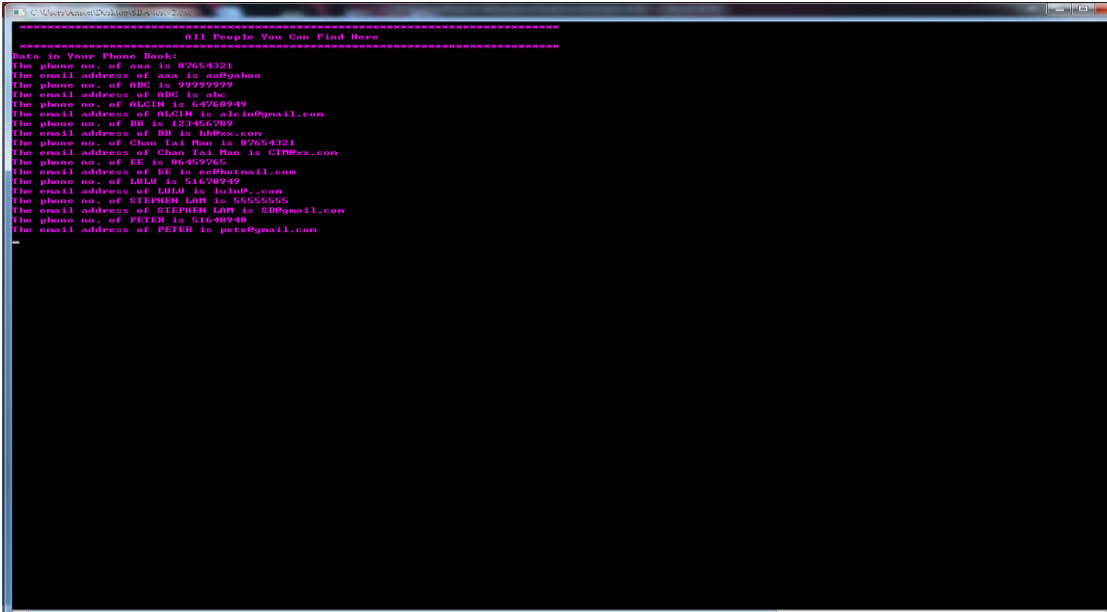


```

C:\Users\Amin\Desktop>python3 1.py
*****
                        Add New Person
*****
Please enter the name of the new record: Peter
Please enter the new phone number: 51648948
Please enter the new email address: peter@gmail.com

```

For adding data, I tested by adding peter and his data.



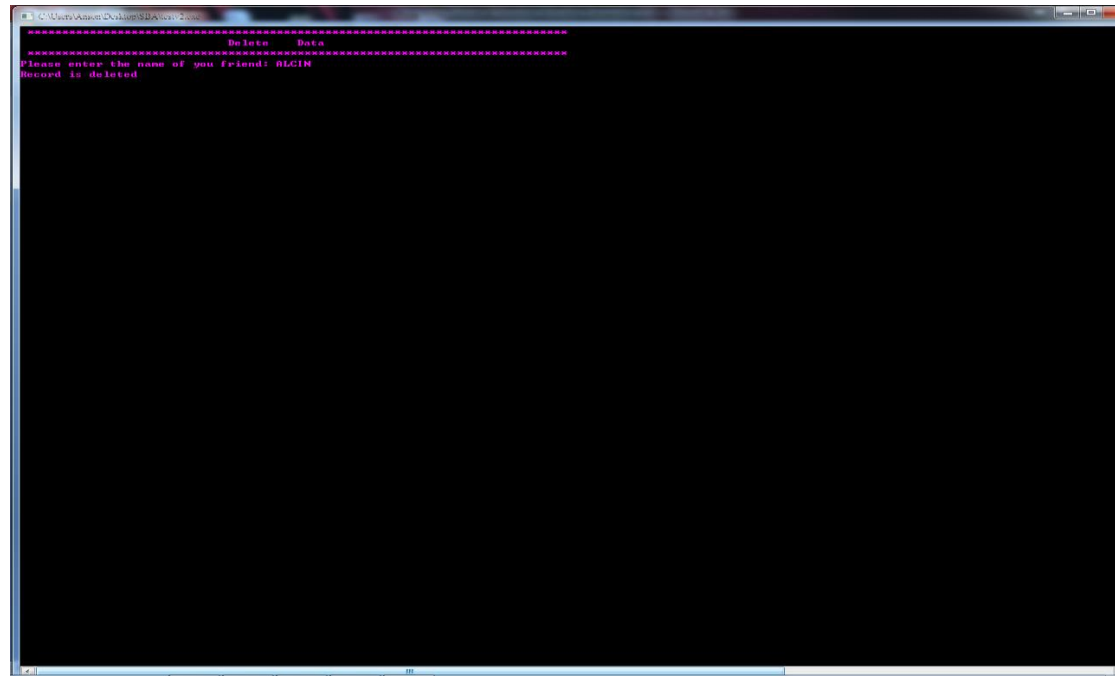
```

C:\Users\Amin\Desktop>python3 1.py
*****
                        All People You Can Find Here
*****
Data in Your Phone Book:
The phone no. of aaa is 82654321
The email address of aaa is aa@yahoo
The phone no. of abc is 99999999
The email address of abc is abc
The phone no. of alicia is 64768949
The email address of alicia is alicia@gmail.com
The phone no. of bb is 123456789
The email address of bb is bb@xx.com
The phone no. of Chan Tai Man is 82654321
The email address of Chan Tai Man is CTM@xx.com
The phone no. of ee is 86459765
The email address of ee is ee@hotmail.com
The phone no. of Lulu is 51678949
The email address of Lulu is lolu@.com
The phone no. of STEPHEN LAM is 55555555
The email address of STEPHEN LAM is SL@xx.com
The phone no. of PETER is 51648948
The email address of PETER is peter@gmail.com

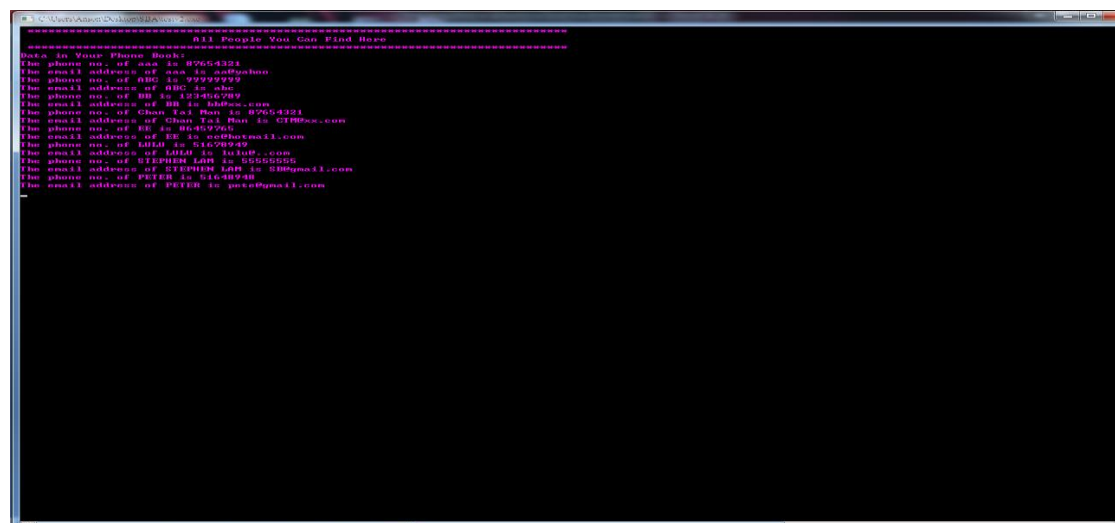
```

Peter's data has been added successfully.

Test 6 : Delete data



I tested by deleting Alcin's data.



Alcin's data disappears.

Conclusion

4.1 Evaluation

Generally, I think that the program has met the minimum requirements of the intended users. The program has made to be user friendly and future use. There are not much complicated source codes and so amendments could be done more easily to meet every users' requirements.

In this program, I have learned more skills on producing a program. Something I have not tried is making different procedures in one main program. And I have also learned that procedures cannot run if it is placed wrongly in the program. I have more to learn in the future. It may helps to my future career.

4.2 Future Plans

- Using a better software for a better interface
- Making more and better functions to improve the quality

4.3 Schedule

Design of Schedule : 10/8/2015 - 20/8/2015

Program Coding: 21/8/2015 - 31/10/2015

Testing and Debugging: 1/11/2015 - 31/11/2015

Writing Report: 1/12/2015 - 11/1/2016

4.4 Appendix

References:

- www.hkeaa.edu.hk
- www.eclass.cswcss.edu.hk
- www.google.com
- <http://en.wikipedia.org>

Thanks to my friend who sit next to me - Stephen Lam

Thanks to my dearest ICT teacher - Mr. Chu Kin Fung

Thanks to my ICT 5** friend - Bobby Chan

Thanks to my computer

Thanks god

Source code

```
program phonebook;
```

```
uses Crt;
```

```
const   Max = 100;  
        datafile = 'Phone.txt';
```

```
var     Name : array[1..Max] of string;  
        Phone : array[1..Max] of string;  
        Email : array[1..Max] of string;  
        N, Choice : integer;
```

```
procedure Readfile;
```

```
var
```

```
    Infile : text;
```

```
    i : integer;
```

```
begin
```

```
    assign(Infile, datafile);
```

```
    reset(Infile);
```

```
    i := 0;
```

```
    while not eof(Infile) do
```

```
        begin
```

```
            i := i + 1;
```

```
            readln(Infile, Name[i]);
```

```
            readln(Infile, Phone[i]);
```

```
            readln(Infile, Email[i])
```

```
        end;
```

```
    close(Infile);
```

```
    N := i
```

```
end;
```

```
procedure WriteFile;
```

```
var Outfile : text;
```

```
    i : integer;
```

```
begin
```

```
    assign(Outfile, datafile);
```

```

rewrite(Outfile);
for i := 1 to N do
    begin
        writeln(Outfile, Name[i]);
        writeln(Outfile, Phone[i]);
        writeln(Outfile, Email[i])
    end;
close(Outfile)
end;

```

```

procedure DisplayData;
var i : integer;
begin
    clrscr;
    writeln('
*****
***** ');
    writeln('                                All People You Can Find Here
');
    writeln('
*****
***** ');
    writeln('Data in Your Phone Book: ');
    for i := 1 to N do
        begin
            write('The phone no. of ');
            write(Name[i]);
            writeln(' is ', Phone[i]);
            write('The email address of ');
            write(Name[i]);
            writeln(' is ', Email[i])
        end
    end;
end;

```

```

procedure sorting; {sort for the DisplayData}
var
    k, i : integer;

```

```

    temp_name, temp_phone, temp_email : string;
    swap : boolean;
begin
    clrscr;
    writeln('
*****
***** ');
    writeln('                                sort data by name
');
    writeln('
*****
***** ');
        k := 0;
    repeat
        k := k + 1;
        swap := false;
        for i := 1 to N - k do
            if upcase(name[i]) > upcase(name[i + 1]) then
                begin
                    temp_name := name[i];
                    name[i] := name[i + 1];
                    name[i + 1] := temp_name;
                    temp_phone := phone[i];
                    phone[i] := phone[i + 1];
                    phone[i + 1] := temp_phone;
                    temp_email := email[i];
                    email[i] := email[i + 1];
                    email[i + 1] := temp_email;
                    swap := true
                end;
            writefile;
        until (k = N - 1) or not swap;
    writeln('Data Sorted!!');
end;

Procedure AddData;
var NewName, NewPhone, NewEmail : string;
    valid : boolean;

```

```

        i : integer;
begin
clrscr;
writeln('
*****
***** ');
writeln('                                Add New Person
');
writeln('
*****
***** ');
repeat
writeln;
    valid := false;
    write('Please enter the name of the new record: ');
    readln(NewName);
    for i := 1 to length(NewName) do
    If (NewName[i] in ['A'..'Z','a'..'z',' ']) then
        valid := true;
        writeln;
    If Not valid then
        write('Invalid Name!');
        writeln
    until valid;

repeat
writeln;
    valid := false;
    write('Please enter the new phone number: ');
    readln(NewPhone);
    for i := 1 to length(NewPhone) do
    If (NewPhone[i] in ['0'..'9',' ']) then
        valid := true;
    If length(NewPhone) > 8 then
        valid := false;
        writeln;
    If not valid then
        write('Invalid Phone Number!');

```

```

writeln;
until valid;
write('Please enter the new email address: ');
readln(NewEmail);
N := N+1;
Name[N] := Upcase(NewName);
Phone[N] := NewPhone;
Email[N] := NewEmail;
writefile;
end;

procedure Search;
var i : integer;
    FindName : string;
    found : boolean;
    ans : char;

begin
clrscr;
writeln('
*****
***** ');
writeln('                                Finding a Person
');
writeln('
*****
***** ');
    writeln('Who do you want to search? : ');
    readln(FindName);
    found := false;

    for i := 1 to N do
    begin

        if upcase(Name[i]) = upcase(FindName) then

            begin

```

```

        found := true;
        write('The phone number of ', FindName, ' is ', Phone[i]);
        writeln;
        write('The email address of ', FindName, ' is ', Email[i]);
        writeln;
    end
end;

```

If NOT found then

```

begin
    writeln('No Data of This Person!!');
    writeln('Enter Y to add record. ');
    readln(ans);
    If (ans = 'Y') or (ans = 'y') then
        AddData;
    end;
end;

```

end;

procedure Changename;

var i : integer;

FindName : string;

Found : boolean;

begin

clrscr;

writeln('

***** ');

writeln('

Making Amendments On Name

');

writeln('

***** ');

writeln('Please enter a name : ');

readln(FindName);

Found := false;

begin

for i := 1 to N do

```

begin
    if upcase(Name[i]) = upcase(FindName) then

begin
    Found := true;
begin
    write('The current name is ', Name[i]);
    writeln;
    write('Enter the new name : ');
    readln(Name[i]);

    end;
    end;
end;
end;
writefile;
end;

procedure Changephone;
var i : integer;
    FindName : string;
    Found : boolean;
begin
    clrscr;
    writeln('
*****
***** ');
    writeln('
                                Making Amendments On Phone
Number                                ');
    writeln('
*****
***** ');
    writeln('Please enter a name : ');
    readln(FindName);
    Found := false;
begin
    for i := 1 to N do
begin

```

```

    if upcase(Name[i]) = upcase(FindName) then

begin
    Found := true;
begin
    write('The phone number of ', FindName, ' is ', Phone[i]);
    writeln;
    write('Enter the new phone number : ');
    readln(Phone[i]);

    end;
    end;
end;
end;
writefile;
end;

procedure Changeemail;
var i : integer;
    FindName : string;
    Found : boolean;
begin
    clrscr;
    writeln('
*****
***** ');
    writeln('
                                Making Amendments On Email
Address                                ');
    writeln('
*****
***** ');
    write('Please enter a name : ');
    readln(FindName);
    Found := false;
begin
    for i := 1 to N do
begin
    if upcase(Name[i]) = upcase(FindName) then

```



```

begin
    Found := true;
begin
    write('The email address of ', FindName, ' is ', Email[i]);
    writeln;
    write('Enter the new email address : ');
    readln(Email[i]);

    end;
end;

end;
end;
writefile;
end;

procedure changedata;
var choice : integer;
begin
repeat
clrscr;
writeln('
*****
***** ');
writeln('
Change      Data
');
writeln('
*****
***** ');
writeln;
writeln('
1. Name Change
');
writeln;
writeln('
2. Phone Number Change
');
writeln;
writeln('
3. Email Address Change
');

```

```

writeln;
writeln('                                4. End
');
writeln;
write('Function: ');
readln(choice);
writeln;
case Choice of
    1:Changename;
    2:Changephone;
    3:Changeemail;
    end;
until choice = 4;
writefile
end;

```

```

procedure DelData;
var i, Locate : integer;
    FindName : string;
begin
clrscr;
writeln('
*****
***** ');
writeln('                                Delete    Data
');
writeln('
*****
***** ');
write('Please enter the name of you friend: ');
    readln(FindName);
    Locate := 0;
    for i := 1 to N do
        if Name[i] = FindName
            then Locate := i;
    if Locate > 0 then
        begin

```

```

        for i := Locate to N-1 do
        begin
            Name[i] := Name[i+1];
            Phone[i] := Phone[i+1];
            Email[i] := Email[i+1]
        end;
        writeln('Record is deleted');
        N := N - 1
    end
    else writeln('No such person');
writeln;
end;

begin

readfile;
repeat
    Clrscr;
    textcolor(13);
writeln('
*****
***** ');
writeln('                                Welcome to The Phone Book
');
writeln('
*****
***** ');
    writeln('1. Display');
    writeln;
    writeln('2. Sorting');
    writeln;
    writeln('3. SEARCH');
    writeln;
    writeln('4. Change Data');
    writeln;
    writeln('5. Add Data');
    writeln;

```

```
writeln('6. Delete Data');
writeln;
writeln('7. End');
writeln;
WRITE('What do you want to do??');
writeln;
write('Function: ');
readln(choice);
writeln;
  case Choice of
    1:DisplayData;
    2:Sorting;
    3:search;
    4:changedata;
    5:AddData;
    6:DelData;
  end;
  readln
  until choice = 7;
  writefile

end.
```

End

Thank You