Project Report
PhoneBook in C

—

Mahmoud EL Banna	6407	Group 4
Ibrahim EL Ghotmy	6281	Group 2
Amr Ashraf	6301	Group 2

Load

This function runs automatically on start and it transfers the data saved in the file (phonebook.txt) to the array of structs in order to be

- It reads the data from the file with a specific order and uses the ',' and the '-' characters as delimiters.
- Every line is a new contact, and every comma is a specific data about this contact.

Menu

- The menu is the home page for our program, from which we can access our different functions by entering one character (1 , 2 , 3 , ... , 8).

```
                      Menu

(1)Search        (2)Add           (3)Delete
(4)Modify        (5)Print         (6)Save
                 (7)Quit

            Choice from 1 to 8:
```

- If the user entered an invalid character the program will request him to re-enter a valid character.

Add

We use this function to add a new contact to our phone book by scanning all the user information :

- The first name
    - We permit the user to enter letters and digits.

- The Last name
    - We permit the user to enter letters and digits.

- The Phone number :
    - We make sure that the user has entered an array of numbers by using the "isdigit" function, and he may also enter a space or the '+' character.

```
Enter the phone number:
abcd
Invalid Phone Number
Enter the phone number:
01007314141
```

- The E-mail :
    - the validation of the email is made by :
        - Checking for the presence of a string before the '@'

        - Checking for the presence of a string between the '@' and the '.'

        - Checking for the presence of a string after the '.'

```
Enter the e-mail:
test@.com
Invalid email
Enter the e-mail:
@gmail.com
Invalid email
Enter the e-mail:
test@gmail.com
```

- Address :
    - We permit the user to enter letters, digits, and spaces.

- Date of birth :
    - We check that the user has entered a digit and not a char by scanning the input of the user as a string, then if it's a digit we save it in an integer using the (sscanf) function.
    - Then we check that this number is valid :
        - for example, there are only 12 months in the year, so if the user entered the number 13 in "month" it's invalid and the program ask him to re-enter the date.

```
Enter the date of birth:
Enter the day:
12
Enter the month:
13
Invalid Date
Enter the month:
12
Enter the year:
1998
```

Delete

- Prompt the user for the last and the first name of the desired contact and the record associated with that name is deleted from the phonebook.

Modify

- Prompt the user for the last name and displays a list of contacts having the same last name for the user to choose from.

```
Enter the last name: jason
(1)Michael       (2)Mark
Enter the number of the person that you want to modify:
```

- After choosing, it asks the user for the type of modification needed.

```
(1)Modify Last name                (2)Modify First Name                (3)Modify Date of Birth
(4)Modify Address                  (5)Modify Email                     (6)Modify Phone Number
```

- And then asks the user to input the new data to replace the old one, and sends the new data to the validation functions, the same as the "Add" function.

```
Enter the new date of birth:
Enter the day:
46
INVALID DATE

Enter the day:
12
Enter the month:
12
Enter the year:
1998
Press any key to continue:
```

- Every time the user inputs any data in any parts of the program these data pass over the "input" and the "validation" functions to ensure that all the data are valid

Print

- The program prompt the user to choose the way of sorting the contacts (Sorting is made by the "Bubble sort" algorithm) :
  - By last name (alphabetically)
    - Using strcasecmp to compare between the names and order them according to their alphabets.

  - By date of birth (from younger to older)
    - Using (< , == , >) to compare between the numbers of the dates, firstly we compare the year, if they are the same we compare the months, if they are the same we compare the day.

Save

- The program transfers the data saved in the array of structs to the file by "w" which remove the data already saved in the file and writes new one.
- We must save the data with the same format in the file which is Lastname,Firstname,day-month-year,Address,email,number.
- And then we close the file after transferring the data by the function "fclose( )"

```
Directory Saved Successfully
Press any key to continue:
```

Quit

- There is a flag put as "true" at the start of the program,
  - Once the user make any change (Add, Modify, Delete) the flag is put to "false".
  - Once the user press on the save function the flag is put to "true".
- If the user press on the quit function without saving the changes, the program will warn the user that the changes aren't saved, but if the user has saved or hasn't made any changes it won't warn him and the program will close.

```
Warning all changes will be Discarded!
(1)Back to Menu              (2)Quit Anyway              (3)Save the changes
```