CSC 215
Procedural Programming with C
Lab #9
Files

Tutorial Section

- In the main method, do the following:
  - Opening a file:
  - **Open the file "hello.txt" in read mode**. To read the file only.
    - `FILE *f = fopen("hello.txt","r");`
    - **WARNING:** If the file doesn't exist, NULL is returned.
    - If you used "r+" instead of "r", the file will open for both reading and writing.
  - **Open the file "output.txt" in write mode**. To create a file and write into it.
    - `FILE *f = fopen("hello.txt","w");`
    - **WARNING:** If a file with the same name already exists its content is **erased** and the file is considered as a new empty file.
    - If you used "w+" instead of "w", an empty file will be created for both reading and writing.
  - **Open the file "output.txt" in append mode**. To add text at the end of the file.
    - `FILE *f = fopen("hello.txt","a");`
    - If the file doesn't exist, it will create the file. And if the file exists it will write the new content after the last character in the file.
    - If you used "a+" instead of "a", the file will open for both reading and appending.
  - **Rewinding a file.** To read a file again from the beginning.
    - `rewind(f);`. Where f is a FILE pointer.
  - **Reading from a file:**
    - Reading a single char. If c == EOF, that means you reached the end of the file.
      - `c = fgetc(f);` Where c is a char and f is a FILE pointer.
    - Reading a string of known length or a line (including the \n). If the method returns NULL, that means you reached the end of the file.
      - `fgets(str, 100, f);` Where str is a string to store the read string into and f is a FILE pointer. The 100 is the maximum number of chars to read.
    - Reading a formatted input. The method returns the number of correct reads.
      - `fscanf(f, "%s %d", str, num);` Where str is a string, num is an integer and f is a FILE pointer. We are reading 2 variables here, so if 2 was returned it means that fscanf was successful.
  - **Writing to a file:**
    - Writing a single char.
      - `fputc(c, f);` Where c is a char and f is a FILE pointer.
    - Writing a string into a file.
      - `fputs(str, f);` Where str is a string to store the read string into and f is a FILE pointer.
    - Writing a formatted input.
      - `fprintf(f, "%s %d", str, num);` Where str is a string, num is an integer and f is a FILE pointer.
  - **Closing a file.** Once you are done with a file, close it.
    - `fclose(f);` Where f is a FILE pointer you want to close.

Lab Section

- Write a program that does the following:
  - Define the constant MAX with the value of 50.
  - Define the constant FileName with your first name as its value.
  - Show the following menu to the user:
    1. Add a New Name.
    2. Display Names.
    3. Delete All Names.
    4. Exit.
  - When the user chooses 1, use ***addName***. The show the menu again.
  - When the user chooses 2, use ***displayNames***. The show the menu again.
  - When the user chooses 3, use ***deleteNames***. The show the menu again.
  - When the user chooses 4, print "Goodbye" then close the program.
  - When the user chooses anything else, show the user an error. The show the menu again.

- Write the following functions:

  - Write the function ***addName*** that takes a string which contains the file name. Read a first and last name from the user. Then append them to the received file name in a single line with a space between them and a new line character at the end.
    - `void addName(char *file);`
  - Write the function ***displayNames*** that takes a string which contains the file name. Read the names line by line then display them. Make sure not to display the new line in the end of the read string.
    - `void displayNames(char *file);`
  - Write the function ***deleteNames*** that takes a string which contains the file name. The function will delete all the names in the file. *(hint: overwrite the file)*
    - `void deleteNames(char *file);`

- Show your program to the instructor. Then upload it to LMS under Lab9 Homework.

SUBMIT POLICY: -

- Use the follow naming convention: Lab09_ID_FirstName_LastName.c
  - **Example:** Lab09_123456789_Marwan_Almaymoni.c
- Use a comment to write your name and ID at the beginning of the code.
- The Deadline is: 29/04/2015 right before the Lab starts.
- Late submissions will not be accepted.
- Email submissions will not be accepted.
- **-1 Point** for not following the naming convention.
- **-1 Point** for not writing your name and ID in the code inside a comment.
- **-8 Points** if the submitted program didn't work due to syntax errors.
- **-10 Points** for cheating and helping others cheat.

Example runs:

```
$ ./lab8
*****************************
* 1- Add a New Name.        *
* 2- Display Names.         *
* 3- Delete all Names.      *
* 4- Exit.                  *
*****************************
> 2
*****************************
No Names were added.
*****************************
* 1- Add a New Name.        *
* 2- Display Names.         *
* 3- Delete all Names.      *
* 4- Exit.                  *
*****************************
> 5
*****************************
ERROR: Incorrect input.
*****************************
* 1- Add a New Name.        *
* 2- Display Names.         *
* 3- Delete all Names.      *
* 4- Exit.                  *
*****************************
> 1
*****************************
Enter the first name: Marwan
Enter the last name: Almaymoni
*****************************
* 1- Add a New Name.        *
* 2- Display Names.         *
* 3- Delete all Names.      *
* 4- Exit.                  *
*****************************
> 1
*****************************
Enter the first name: Ali
Enter the last name: Baba
*****************************
* 1- Add a New Name.        *
* 2- Display Names.         *
* 3- Delete all Names.      *
* 4- Exit.                  *
*****************************
> 2
*****************************
1- Marwan Almaymoni.
2- Ali Baba.
*****************************
* 1- Add a New Name.        *
* 2- Display Names.         *
* 3- Delete all Names.      *
* 4- Exit.                  *
*****************************
> 3
*****************************
Delete is Complete.
```

```
****************************
* 1- Add a New Name.        *
* 2- Display Names.         *
* 3- Delete all Names.      *
* 4- Exit.                  *
****************************
> 2
****************************
No Names were added.
****************************
* 1- Add a New Name.        *
* 2- Display Names.         *
* 3- Delete all Names.      *
* 4- Exit.                  *
****************************
> 4
****************************
Goodbye!
```