Software Engineering Project 1

(Comp 10050)

Assignment 1 – the iPod Music player

Aim: to create a program that sort and shuffle a list of songs received as input

You are expected to perform this assignment individually

A. Detailed Specification

The objective for this assignment is to create a program that:

1) Asks the user to provide as input from the command line an artist name. At most 4 artists can be provided as input. For each artist, at least 1 and at most 3 songs could be provided. To indicate you do not want to add more artists or more songs, you can press "Enter" or a character of your choice.

For example, this is how the console should ask the users to provide the singers/groups names and their songs as input. Note that command line output is indicated in black while the command line input is indicated in blu.

```
Insert an artist/group name:
Rihanna

Insert song 1 for Rihanna
Work
Insert song 2 for Rihanna
Diamonds
Insert song 3 for Rihanna
Umbrella

Insert an artist/group name:
The Cranberries

Insert song 1 for The Cranberries
Zombie
Insert song 2 for The Cranberries
Salvation
Insert song 3 for The Cranberries
Promises

Insert an artist/group name:
```

```
U2

Insert song 1 for U2
With or Without you
Insert song 2 for U2
One
Insert song 3 for U2
Beautiful Day

Insert an artist/group name:
Kendrick Lamar

Insert song 1 for Kendrick Lamar
HUMBLE.
Insert song 2 for Kendrick Lamar
DNA.
Insert song 3 for Kendrick Lamar
LOYALTY.FEAT.RIHANNA.
```

2) Then the program should sort and print the list of songs given as input depending on the artist/group name and the song title.

For example, this should be the output for the list of artists/group names and songs provided at step 2.

```
Sorted list of songs:
Kendrick Lamar
    -   DNA.
    -   HUMBLE.
    -   LOYALTY.FEAT.RIHANNA.

Rihanna
    -   Diamonds
    -   Umbrella
    -   Work

The Cranberries
    -   Promises
    -   Salvation
    -   Zombie

U2
    -   Beautiful Day
    -   One
    -   With or Without you
```

3) Create a random playlist of the songs given as input ensuring that the same song can only appear again after at list 5 different other songs have played. The same song must appear twice.

For example, this is a possible output for the list of artists/group names and songs provided at step 2.

```
Shuffled Playlist:
Kendrick Lamar - HUMBLE.
The Cranberries - Zombie
The Cranberries - Promises
Rihanna - Diamonds
Rihanna - Umbrella
U2 - One
Kendrick Lamar - HUMBLE.
U2 - With or Without you
The Cranberries - Salvation
Rihanna - Work
U2 - Beautiful Day
Rihanna - Diamonds
The Cranberries - Promises
Rihanna - Umbrella
U2 - One
Kendrick Lamar - LOYALTY.FEAT.RIHANNA.
The Cranberries - Salvation
Kendrick Lamar - DNA.
The Cranberries - Zombie
Rihanna - Work
U2 - Beautiful Day
Kendrick Lamar - LOYALTY.FEAT.RIHANNA.
U2 - With or Without you
Kendrick Lamar - DNA.
```

Requirements:

1. To sort the players use any algorithm different from the bubble sort.
2. To shuffle the players you can use the Fisher Yates algorithm
3. Your programme should work for other list of artists/groups and songs.

Code Design Requirements:

- Comment your code,
- Use functions where you can.
- Separate your code into independent modules

Design Hints:

1. Start by getting the input of the playlist working

2. Remember that strings cannot be the target of an assignment (=), you have to use strcpy.
4. Strings cannot be compared using == or < so you need to use strcmp or strncmp.
5. For shuffling consider the Fisher Yates algorithm: https://en.wikipedia.org/wiki/Fisher–Yates_shuffle

```
To shuffle an array a of n elements (indices 0..n-1):
 for i from n − 1 downto 1 do
       j ← random integer such that $0 \le j \le i$
        exchange a[j] and a[i]
```

6. The Fisher Yates algorithm will not meet the requirement about not having consecutive songs so that you will have to handle it separately. Get shuffle working ignoring this requirement initially.

Google strcpy, strcmp, strncmp, and strstr. Get used to one of the standard online resources, e.g.
https://en.wikibooks.org/wiki/C_Programming/C_Reference/string.h/strcpy

Submission:

- Submit two items through Moodle,
    - an archive file (e.g., .zip or .tar.gz) containing your source module
    - a text file (.doc or .pdf) providing implementation details and comments on the design decisions you have made.
        - What function did you use to read the input from the command line? Why?
        - What algorithm did you choose to sort the playlist? Why?
        - How did you implement the shuffling algorithm?

Evaluation Criteria

A mark [0-100] will be give according to the following criteria

- The code is well commented and appropriately divided into modules (10 points)
- Input from command line is read correctly. (15 points)
- The submitted text file describes your design choices appropriately (5 points)
- Sorting algorithm is correct and has good performance (30)
- Shuffling algorithm is randomized and satisfied the criteria described in part A.3 in this document (40 points)