Coding Assignment 4

You are creating a game called "00000011 STRIKES – YOU'RE OUT – the CSE version". It is a version of the classic game Hangman and uses songs for the phrases to be guessed. The first round of the game is guessing the song's first lyric. The player will choose from a menu of 5 songs. Based on the menu choice, a song lyric will be displayed with all letters replaced with dashes. Any numbers or punctuation will remain. The player will enter guesses as individual letters. If the player guesses a correct letter, then the phrase will be redisplayed with the missing letter filled in. If the user guesses wrong, they will earn a strike. Strikes will be display in binary since this is the CSE version. If the player earns three strikes, then the game is over and the player has lost.

Rules of the Game

1. A player's guess must replace both the uppercase and lowercase versons of the guessed lettter.
2. All phrases to be guessed will contain upper and lowercase letters and will be displayed with those letters.
3. When the user guesses an incorrect letter, the number of strikes will be displayed in binary. Guessing punctuation or numbers counts as an incorrect letter/guess and earns the player a strike.
4. If the player wins Round 1 (song lyric), then the game asks if the player wants to play Round 2 (the song title).
5. If the player wins Round 2 (song title), then the game asks if the player want to play Round 3 (the song's artist).
6. The number of strikes should be reset after each round.

Step 1 – Alter your library files - `MyLib.c` and `MyLib.h` file.

1. `ConvertDecimalToBinary()` should only take one parameter now –the decimal number to be converted
2. `ConvertDecimalToBinary()` should convert the decimal number to binary and print it (move the code from `PrintBinary()` into `ConvertDecimalToBinary()`).
3. Eliminate `PrintBinary()` now that `ConvertDecimalToBinary()` prints the binary number.

Part 2 – `songlist.txt`

The provided file "`songlist.txt`" consists of 5 variables of type `SONG`.

```
SONG song1 = {"Bangles","Walk Like An Egyptian","All the old paintings on the tombs"};

SONG song2 = {"Heart","Alone","I hear the tickin' of the clock"};

SONG song3 = {"Gregory Abbott","Shake You Down","Girl, I been watching you"};

SONG song4 = {"Whitney Houston","I Wanna Dance With Somebody","Clock strikes upon the hour"};

SONG song5 = {"Starship","Nothing's Gonna Stop Us Now","Looking in your eyes I see a paradise"};
```

You will need to create a structure typedefed to `SONG`. The structure will consist of 3 character pointers to `artist`, `title` and `firstline`. You will need to create and submit your own "`songlist.txt`" file and submit it with your code.

Step 3 – In `main()` in your `Code4.c` file

The first line in your main function will be

```
#include "songlist.txt"
```

This will create 5 SONG variables for you. For your initial creation and testing, please use the "songlist.txt" attached to the Blackboard assignment. You will be submitting a file of your own with your assignment and the GTA's will be testing using both your file and a file of my creation. This will ensure that no assumptions (hardcoding) is done for string lengths or content.

Create a songarray of type SONG and assign the addresses of song1, song2, song3, song4 and song5 to it.

Print out the game's menu and accept the player's song choice. Use the player's song choice to retrieve the song's title from the songarray. Use a loop to take the player through Round 1 – guessing the song's lyrics, Round 2 – guessing the song's title and Round 3 – guessing the song's artist. The player should only continue to the next round if they win the current round.

Part 4 – makefile

Change your makefile to use your new Code4.c file. You are using the same name for the library so you should not need to change that in the makefile. Compile your program and run your program. Remember how the makefile should be named to work on Omega.

Part 5 - Testing

Run your Code4.e and confirm that your output matches the output in the assignment. Confirm that you have met all elements of the rubric and your game follows the rules of the game.

Part 6 – Code Submission

Submit a zip fle containing the following files

Code4_xxxxxxxxxx.c

MyLib.c

MyLib.h

makefile

songlist.txt (your version)

Coding Hints

Make a copy of your phrase and uppercase that copy. Use that uppercase version for searching for the uppercased version of the player's guess. That way you will catch both upper and lower case without having to check for both.

Use strpbrk() with the entire uppercased alphabet to replace only characters (not numbers or punctuation) in the phrase with dashes.

Your SONG typedefed structure is using only pointers so that the song lyric, title and artist can be any length.

Output From Runs of `Code4.c`

```
Welcome to 00000011 STRIKES - YOU'RE OUT - the CSE version


Pick a song


0. Exit
1. Song 1
2. Song 2
3. Song 3
4. Song 4
5. Song 5
Song Choice 1


Round 1 - Here's the song lyric you need to guess



--- --- --- --------- -- --- -----


Player : Guess a letter : a



A-- --- --- -a------- -- --- -----


Player : Guess a letter : e



A-- --e --- -a------- -- --e -----


Player : Guess a letter : i



A-- --e --- -ai--i--- -- --e -----


Player : Guess a letter : o
```

```
A-- --e o-- -ai--i--- o- --e -o---

Player : Guess a letter : u

Strike 00000001

A-- --e o-- -ai--i--- o- --e -o---

Player : Guess a letter : n


A-- --e o-- -ain-in-- on --e -o---

Player : Guess a letter : t


A-- t-e o-- -aintin-- on t-e to---

Player : Guess a letter : h


A-- the o-- -aintin-- on the to---

Player : Guess a letter : l


All the ol- -aintin-- on the to---

Player : Guess a letter : d


All the old -aintin-- on the to---
```

```
Player : Guess a letter : g



All the old -ainting- on the to---

Player : Guess a letter : s



All the old -aintings on the to--s

Player : Guess a letter : p



All the old paintings on the to--s

Player : Guess a letter : m



All the old paintings on the tom-s

Player : Guess a letter : b

You guessed correctly!!

All the old paintings on the tombs

Congratulations!  You guessed correctly - ready for Bonus Round 2?

Do you want to play the next round? y

Round 3 - Here's the artist you need to guess



-------
```

```
Player : Guess a letter : a


-a-----

Player : Guess a letter : e


-a---e-

Player : Guess a letter : i

Strike 00000001

-a---e-

Player : Guess a letter : o

Strike 00000010

-a---e-

Player : Guess a letter : u

Strike 00000011

00000011 STRIKES - YOU'RE OUT!!

Game over
```

```
Welcome to 00000011 STRIKES - YOU'RE OUT - the CSE version

Pick a song

0. Exit
1. Song 1
2. Song 2
3. Song 3
4. Song 4
5. Song 5
Song Choice 5

Round 1 - Here's the song lyric you need to guess

------- -- ---- ---- - --- - --------

Player : Guess a letter : a

------- -- ---- ---- - --- a -a-a----

Player : Guess a letter : e

------- -- ---- e-e- - -ee a -a-a---e

Player : Guess a letter : i

----i-- i- ---- e-e- I -ee a -a-a-i-e

Player : Guess a letter : o
```

```
-oo-i-- i- -o-- e-e- I -ee a -a-a-i-e

Player : Guess a letter : u



-oo-i-- i- -ou- e-e- I -ee a -a-a-i-e

Player : Guess a letter : n



-oo-in- in -ou- e-e- I -ee a -a-a-i-e

Player : Guess a letter : l



Loo-in- in -ou- e-e- I -ee a -a-a-i-e

Player : Guess a letter : g



Loo-ing in -ou- e-e- I -ee a -a-a-i-e

Player : Guess a letter : s



Loo-ing in -ou- e-es I see a -a-a-ise

Player : Guess a letter : k



Looking in -ou- e-es I see a -a-a-ise

Player : Guess a letter : q

Strike 00000001
```

```
Looking in -ou- e-es I see a -a-a-ise

Player : Guess a letter : o



Looking in -ou- e-es I see a -a-a-ise

Player : Guess a letter : m

Strike 00000010

Looking in -ou- e-es I see a -a-a-ise

Player : Guess a letter : t

Strike 00000011

00000011 STRIKES - YOU'RE OUT!!

Game over
```