ECE 325 Course Assignment 2: Building a collection of your songs

Now that you have a great band name, it is time to look at your collection of songs, which is growing steadily. You have a list of all your songs in the songs.txt file. As you can tell, song titles are quite peculiar in the neoclassical jazzhopmetal scene – they are all variations of song titles from the greatest music era in history (the 90s). One problem with the list is that you accidentally entered some song titles multiple times, so it is hard to know exactly how many songs you currently have.

You plan to keep track of the list of songs in a SongCollection object, but you have a hard time deciding whether to use an array or ArrayList as the collection type that is used inside the SongCollection class. Since you are a great Java programmer you decide to implement both versions and then decide which one is best.

1) **Implement two versions of the SongCollection class**: SongCollectionArray and SongCollectionArrayList, which are basically wrappers for an array and an ArrayList of String objects. We already provided the class outline in the .ZIP file. All you have to do is finish the classes. Make sure that the collection can only contain unique titles.

2) **In the main function of each class**, read the songs from the songs.txt file and print the number of unique songs in the collection. If you want, you can use (parts of) the readTxt() method from lab assignment 2.

Answer the following questions and include your answers in a .PDF file.

3) **Which class is the better option? Give at least two reasons for why you think this.**

4) **In the remove() method in the SongCollectionArray class, we have the requirement that all empty slots of the array need to be at the end. Why is this important?**

Hints

- Some of the method bodies can be very short.

- Make sure to think about what should happen when someone does not use your class as you intended. For example, what should happen when someone tries to get a song from a negative index? Hint; the comments in the code will tell you what should happen – make sure to implement them closely.

- Make sure not to change any of the method signatures. You are free to add any methods or fields that you deem necessary to finish the classes.
- You can assume that there are never more than 100 songs in a collection.
- The ArrayList class is parameterized in Java 17. We will talk more about what this means later, but for now it is sufficient to know that you can create an ArrayList of String objects as follows:
ArrayList<String> songs = new ArrayList<String>();

Rubric

Implementation of the two classes 14 points total.

Questions 3 points each.

(20 points total)

Please submit:

1) A zip file containing your code and a PDF with the answers to the questions above.
Name the file 'FirstName_ID_course_asg2.zip' and keep the exact same file structure as the zip that was provided for the assignment. So, for example:
Filename: Cor-Paul_1234567_course_asg2.zip
|------- solution.pdf
|------- songs.txt
|------- src
| |------- ece325
| | |------- lec
| | | |------- assignment2
| | | | |------- *.java

2) A screencast/movie that shows the following steps:

- Open your eClass with your name shown
- Open your IDE
- Show your code briefly
- Execute your code and demonstrate that your class works correctly.

Please submit the screencast as a **separate** file to eClass.

Please do not modify any of the names/methods we've defined in the provided *.java files.