Due Date: 11.11.2018 23.59

Homework 2
CENG211 – Programming Fundamentals

In this homework you are expected to implement an "***Orchestra System***" in Java according to the given scenario.

You should fulfill the concepts of;

- Iterator
- Inheritance
- Polymorphism

You are given four files that consists of capital letters and numbers such as A1, B0.25, C0.5 etc. These letters represent **musical score**s and correspond to Turkish musical score symbols DO, RE etc. Numbers represent the beat of the score given next to it. For example, B0.25 means, play the score B in 0.25 beat. Each line in these files represents a different **part** of the **piece**. Scores and parts are played up to their sequence in the file.

The orchestra is directed by a **Maestro** who is responsible of:

- Set ***the tempo*** of the parts according to **the total number of beats in a *part*** in the given piece:
  1- Number of beats < 8, tempo is *Prestissimo* (very very fast)
  2- 8 ≤ Number of beats < 16, tempo is *Vivace* (lively and fast)
  3- 16 ≤ Number of beats < 18, tempo is *Allegretto* (fast, quickly and bright)
  4- 18 ≤ Number of beats < 22, tempo is *Moderato* (moderately)
  5- 22 ≤ Number of beats < 23, tempo is *Adagietto* (rather slow)
  6- 23 ≤ Number of beats < 24, tempo is *Andante* (at a walking pace)
  7- 24 ≤ Number of beats < 27, tempo is *Larghetto* (broadly)
  8- 27 ≤ Number of beats < 29, tempo is *Lento* (slowly)
  9- 29 ≤ Number of beats <33, tempo is *Grave* (slow and solemn)
  10- 33 ≤Number of beats < 37, tempo is *Larghissimo* (very, very slow)
- Set ***the change in tempo*** according to **the total number of beats in a *piece***:
  1- Number of beats < 83, tempo is *Lentando* (gradual slowing and softer)
  2- 83 ≤ Number of beats < 125, tempo is *Ritenuto* (slightly slower)
  3- 125 ≤ Number of beats < 132, tempo is *Stretto* (in faster tempo)
  4- 132 ≤ Number of beats < 152, tempo is *Accelerando* (speeding up)

In this system there will also be an orchestra of **musicians**. The orchestra consists of **string instrument musicians**, **woodwind instrument musicians** and **percussion instrument musicians**. String instrument musicians play **violin**, **viola** and **cello**. Woodwind instrument musicians play **flute**. Percussion instrument musicians play **drum** and **bell**. These musicians play their instruments according to the piece and the tempo given by the maestro.

- String instrument musicians play all parts of a piece

- Woodwind instrument musicians play only chorus
- Percussion instrument musicians play only the last two parts
- Violinists don’t play if the tempo is Prestissimo
- Violists play only the first part of the piece when the change in tempo is Ritenuto
- Flutists play only chorus parts
- Drummers play if the given tempo of the part is one of Prestissimo, Vivace or Allegretto. In addition, Drummers are only able to play the notes C, D and E.
- Bell is played when a part is played at tempo Grave and the change in tempo is Stretto. Furthermore, Bell could only play the notes F, G, A and B.

Each musician should print an output when they play. The printed info should include the change in tempo during playing, the parts of piece and their corresponding tempo and the musician that are currently playing it. For Drummers and Bell players, a note is printed as “*X*” if it is not played. For example:

```
Piece 1 is played Lentando
Violin is played:
Part 1: A B C D G H E E… Allegro
Cello is played:
Part 1: A B C D G H E E… Allegro
…
Flute is played:
Part 3: B C G G D D… Moderato
Bell is played:
Part 3: B X G G X X… Stretto

….
```

In this system, there should be a **Concert** class that is responsible of creating the Pieces, the Maestro and the ArrayList of Musicians (*init()*). Also, there should be a **ConcertApp** that creates a Concert object and starts it. File I/O operations must be done in a different class.

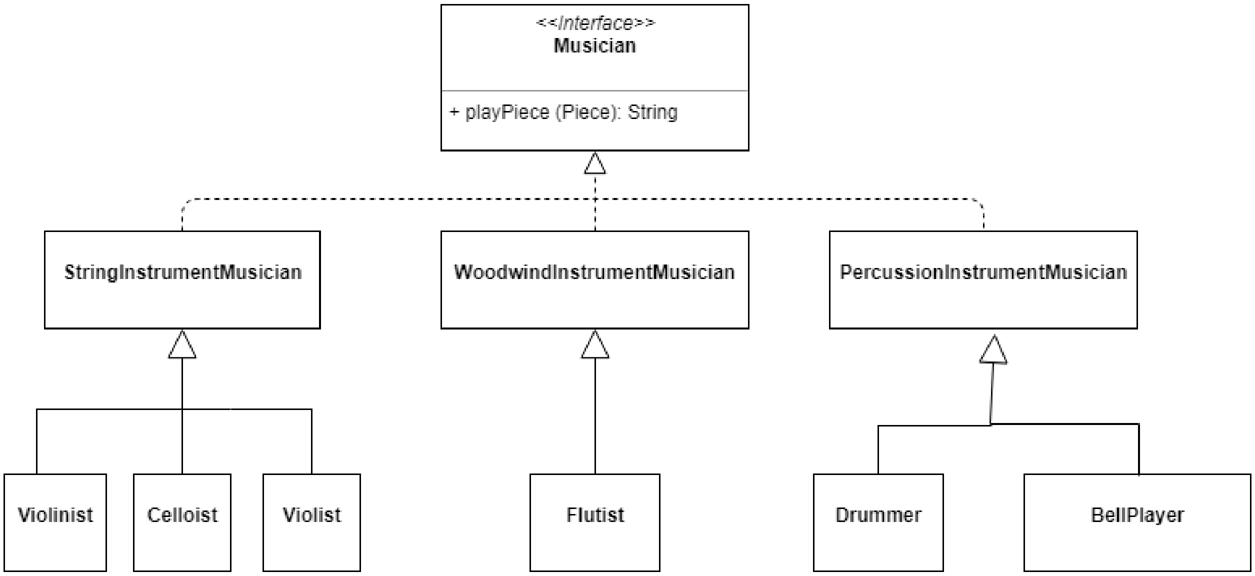
Hint: Chorus is the part that is replayed at least two times.

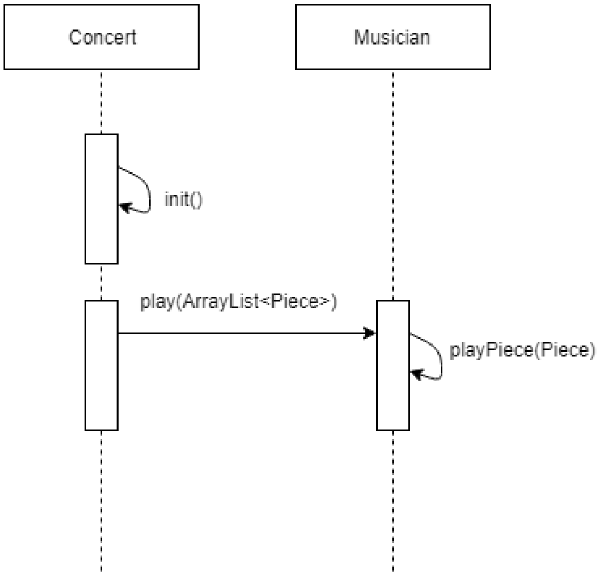
Assignment Rules

1. In this lecture’s homework, there are no cheating allowed. If any cheating has been detected, they will be graded with 0 and there will be no further discussion on this.
2. You are expected to submit your homework in groups. Therefore, only one of you will be sufficient to submit your homework.
3. Make sure you export your homework as Eclipse projects. You can use other IDEs as well, however, you must test if it supported by Eclipse.
4. **Make sure that your “.txt” files are in your project after you exported it.**
5. Please submit your homework through CMS.
6. Please export your Java Project as the given format with your assigned group ID.
   <u>Example:</u>

   G2_CENG211_HW2.zip. (Your group IDs will be announced on CMS).

7. Please be informed that your submissions may be anonymously used in software testing and maintenance research studies. Your names and student IDs will be replaced with non-identifying strings. If you do not want your submissions to be used in research studies, please inform the instructor (Dr. Tuglular) via e-mail.





Hint: You can use the UML class and sequence diagrams to design your classes.