

```

Guitar
strings : int
chords : int
song : double[11]

+ Guitar (strings: int, chords: int)
+ simulateSong(): void
+ getChordsAsArray(chord: int): double[]
+ getSongLength(): int
+ generateSong(): void

```

```

Vocalist
center: double
melody: double[11]
guitar: Guitar

```

```

+ Vocalist (guitar: Guitar, center: double)
+ sing(): void
+ toString(): String
+ generateMelody(): void

```

```

SongWriter
+ main (args: String[]): void

```

```

StdAudio
+ PlayTone (hz: double, duration: double): void

```

← Class name

← Column = chord, Entry = string, last row is beats row

← Class constructor

← Simulate song by printing chord and wait the appropriate

← Returns an array of strings (doubles) from song array.

← Returns the length of the song array (# of columns)

← Creates the song array using random doubles

← Class name

← row index 0 = frequencies, row index 1 = durations.

← Class constructor

← Uses StdAudio to play the melody array.

← Returns a string representation of the melody array.

← Creates the melody array, by searching each chord of the guitar's song array, for the most similar string to the center frequency

← main method where all the magic happens.

← Class name, given by professor

← Plays the hz frequency for the duration time through the computer's speakers.