CodeHS

AP Java FRQ: Combination Lock Game

Directions: SHOW ALL YOUR WORK. REMEMBER THAT PROGRAM SEGMENTS ARE TO BE WRITTEN IN Java.

Notes:

- Assume that the classes listed in the Quick Reference have been imported where needed.
- Unless otherwise noted in the question, assume that parameters in method calls are not `null` and that methods are called only when their preconditions are satisfied.
- In writing solutions for each question, you may use any of the accessible methods that are listed in classes defined in that question. Writing significant amounts of code that can be replaced by a call to one of these methods may not receive full credit.

Consider a game with a combination lock box that has a 4-letter word as the combination. A player tries to guess the combination by guessing one letter at a time to win a prize inside the box. The combination only contains 4 lower-case letters and a guess can only contain 4 lower-case letters.

For each round of play, the player is given a clue based on a comparison between the combination lock and the guess. Each position in the clue contains a character that corresponds to the letter in the same position in the guess. The following rules determine the characters that appear in the clue.

If the letter in the guess is...	**the corresponding character in the clue is**
also in the same position in the combination lock.	the matching letter
also in the combination lock,	`"+"`
not in the combination lock,	`"*"`

The `CombinationLock` class will be used to represent the combination lock in the game. The combination lock is passed to the constructor. The class contains a method, `getClue`, that takes a guess and produces a clue.

For example, suppose the variable `comboLock` is declared as follows:

```
CombinationLock comboLock = new CombinationLock("frog");
```

The following table shows several guesses and hints that would be produced.

Call to getClue	String returned
`comboLock.getClue("oooo")`	`"++o+"`
`comboLock.getClue("flip")`	`"f***"`
`comboLock.getClue("form")`	`"f++*"`
`comboLock.getClue("frag")`	`"fr*g"`
`comboLock.getClue("frog")`	`"frog"`

Write the complete `CombinationLock` class, including any required instance variables, its constructor, and the method `getClue`, described above.