

1) This question involves the creation and use of a spinner to generate random numbers in a game. A `GameSpinner` object represents a spinner with a given number of sectors, all equal in size. The `GameSpinner` class supports the following behaviors.

- Creating a new spinner with a specified number of sectors
- Spinning a spinner and reporting the result
- Reporting the length of the current run, the number of consecutive spins that are the same as the most recent spin

The following table contains a sample code execution sequence and the corresponding results.

Statements	Value Returned (blank if no value is returned)	Comment
<code>GameSpinner g = new GameSpinner(4);</code>		Creates a new spinner with four sectors.
<code>g.currentRun()</code>	0	Returns the length of the current run. The length of the current run is initially 0 because no spins have occurred.
<code>g.spin();</code>	3	Returns a random integer between 1 and 4, inclusive. In this case, 3 is returned.
<code>g.currentRun();</code>	1	The length of the current run is 1 because there has been one spin of 3 so far.
<code>g.spin();</code>	3	Returns a random integer between 1 and 4, inclusive. In this case, 3 is returned.
<code>g.currentRun();</code>	2	The length of the current run is 2 because there have been two 3s in a row.
<code>g.spin();</code>	4	Returns a random integer between 1 and 4, inclusive. In this case, 4 is returned.
<code>g.currentRun();</code>	1	The length of the current run is 1 because the spin of 4 is different from the value of the spin in the previous run of two 3s.
<code>g.spin();</code>	3	Returns a random integer between 1 and 4, inclusive. In this case, 3 is returned.
<code>g.currentRun();</code>	1	The length of the current run is 1 because the spin of 3 is different from the value of the spin in the previous run of one 4.
<code>g.spin();</code>	1	Returns a random integer between 1 and 4, inclusive. In this case, 1 is returned.
<code>g.spin();</code>	1	Returns a random integer between 1 and 4, inclusive. In this case, 1 is returned.
<code>g.spin();</code>	1	Returns a random integer between 1 and 4, inclusive. In this case, 1 is returned.
<code>g.currentRun();</code>	3	The length of the current run is 3 because there have been three consecutive 1s since the previous run of one 3.

Write the complete `GameSpinner` class. Your implementation must meet all specifications and conform to the example.

2) Released AP Exam FRQ – 2015 #2 – Guessing Game

Consider a guessing game in which a player tries to guess a hidden word. The hidden word contains only capital letters and has a length known to the player. A guess contains only capital letters and has the same length as the hidden word.

After a guess is made, the player is given a hint that is based on a comparison between the hidden word and the guess. Each position in the hint 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 hint.

If the letter in the guess is ...	the corresponding character in the hint is
also in the same position in the hidden word,	the matching letter
also in the hidden word, but in a different position,	" + "
not in the hidden word,	" * "

The `HiddenWord` class will be used to represent the hidden word in the game. The hidden word is passed to the constructor. The class contains a method, `getHint`, that takes a guess and produces a hint.

For example, suppose the variable `puzzle` is declared as follows.

```
HiddenWord puzzle = new HiddenWord("HARPS");
```

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

Call to <code>getHint</code>	String returned
<code>puzzle.getHint("AAAAA")</code>	" +A+++ "
<code>puzzle.getHint("HELLO")</code>	"H***** "
<code>puzzle.getHint("HEART")</code>	"H*++* "
<code>puzzle.getHint("HARMS")</code>	"HAR*S "
<code>puzzle.getHint("HARPS")</code>	"HARPS "

Write the complete `HiddenWord` class, including any necessary instance variables, its constructor, and the method, `getHint`, described above. You may assume that the length of the guess is the same as the length of the hidden word.