Progression Classes

Background

Progressions (or Sequences and Series) are numbers arranged in a particular order such that they form a predictable order. By predictable order, we mean that given some numbers, we can find next numbers in the series.

Arithmetic Progression (AP)

A sequence of numbers is called an arithmetic progression if the difference between any two consecutive terms is always same. In simple terms, it means that next number in the series is calculated by adding a fixed number to the previous number in the series. This fixed number is called the common difference.

For example, 2,4,6,8,10 is an AP because difference between any two consecutive terms in the series (common difference) is same (4 – 2 = 6 – 4 = 8 – 6 = 10 – 8 = 2).

If ‘a’ is the first term and ‘d’ is the common difference,

- nth term of an AP = a + (n-1) d

Fibonacci Progression (FP)

A sequence of numbers called an fibonacci progression where a number is found by adding up the two numbers before it. Starting with 0 and 1, the sequence goes 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, and so forth. Written as a rule, the expression is:

- Nth term of FP (x_n) = $x_{n-1} + x_{n-2}$
- $X_0 = 0$, $x_1 = 1$

Your task

I've already defined a Progression class in Progression.java, which can print a sequence of numbers starting with 0 and the next number is the previous number plus one. To test it, A ProgressionTester class is provided in ProgressionTester.java, which uses the Progression class to create an object and use that object to print out the first 10 numbers in the sequence.

When we run the tester, we would get the output as follow:

```
$ java progressionTester
Default progression
0 1 2 3 4 5 6 7 8 9
```

Please use the Progression class as an example to create an **ArithProgression class** that can print out arithmetic progression as described in the background. The class should allow user to specify the first number and the difference between consecutive numbers in the sequence in the class constructor (Hint: you can add two more member variables to implement this, one for the first number, another for the increment).

Also, use the Progression class as an example to create a **FibProgression class** that can print out fibonacci progression as described in the background. The class should allow user to specify the first and second numbers in the sequence in the class constructor (Hint: you can add two more member variables to implement this, one for the first number, another for the second number).

In general you need to add more necessary member variables and re-implement the helper method nextValue() in your own classes. Finally, add statements in **ProgressionTester class** to test the ArithProgression and FibProgression classes by printing the first 10 numbers in these sequences. The expected test cases and output should be as follow:

```
$ java progressionTester
Default progression
0 1 2 3 4 5 6 7 8 9
Arithmetic progression with start value 2 and increment 5:
2 7 12 17 22 27 32 37 42 47
Fibonacci progression with start values 4 and 6:
4 6 10 16 26 42 68 110 178 288
```