VectorOperations

You are given a Java project which includes three files: *Signal.java, CreateAndTestSignals.java* and *AdvancedSignal.java.*

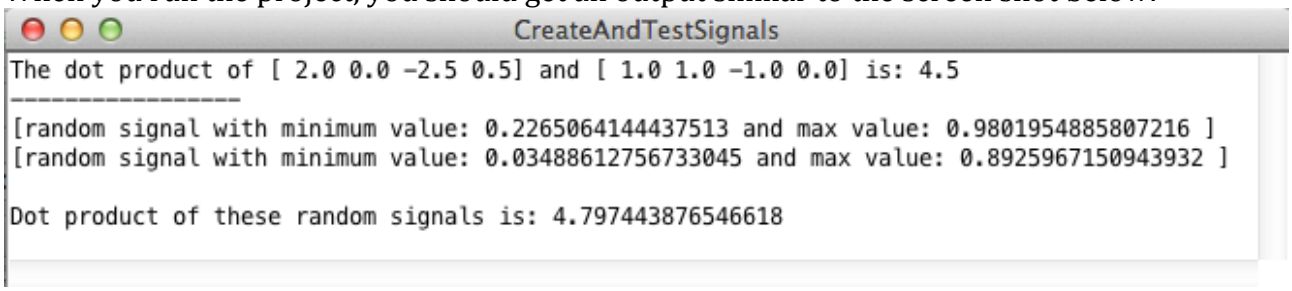
CreateAndTestSignals.java is the class that includes the run method and that is already implemented. As you would see in the implementation, *CreateAndTestSignals* creates two 1x4 vectors, two 1x20 vectors and prints the contents of the vectors and result of their dot-product.

The *Signal* class has an instance variable that is a vector of doubles and has a method named dotProduct which is not implemented. **Your first task is to implement the dotProduct method.** .dotProduct first multiplies the two vectors element by element, than sums up all the resulting values. *Ex*: dotProduct of [2 5] and [-1 3] will be performed by the operation: 2*(-1)+5*3=13. This is similar to dot product in vector calculus with the only difference that the second vector does not need to be a column vector.

Your second task is to write the *AdvancedSignal* class.
The *AdvancedSignal* class should have two additional instance variables: min and max. You should implement getter and setter methods for these new variables as well as findMax() and findMin() methods that should find the min and max value and assign to the variables.

AdvancedSignal should include two constructors: one should accept a double-array (double[] array) as the super class, the second one should accept the length of the array (int size) and create an array carrying random values which will serve as the data. You should also implement a toString method for *AdvancedSignal*, which prints the max and min values as shown in the example below.

When you run the project, you should get an output similar to the screen shot below:



Since random signals will carry different values in each run, you should not expect to get exactly the same numbers but they should be to some level similar.

Hint:
For creating a random generator you can use:
static RandomGenerator rgen=RandomGenerator.getInstance();

For accessing the contents of the private variable *data*, you can use the public methods *getDataAtIndex*() and *getSignalLength*().