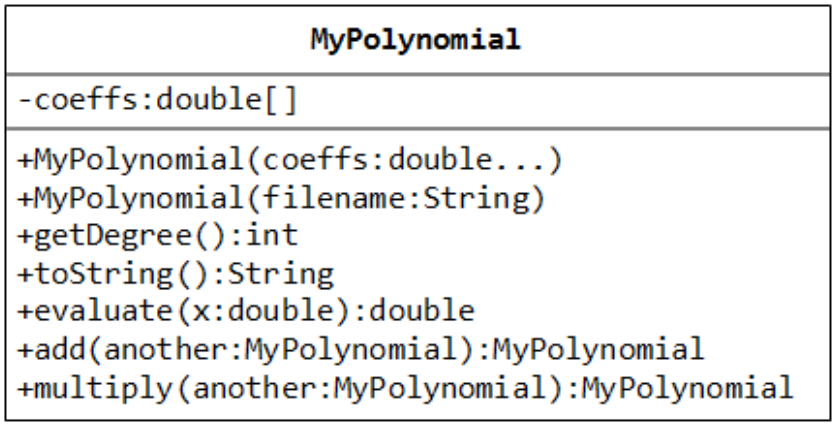
Creating a Polynomial - Exercise

Write a class called `MyPolynomial`, which models polynomials of degree n:
$c_0 + c_1x^1 + c_2x^2 + \ldots + c_nx^n$,

designed as shown in the class diagram.



The class contains:

- An instance variable named `coeffs`, which stores the coefficients of the n-degree polynomial in a `double` array of size `n+1`, where c_0 is kept at index 0.
- A constructor `MyPolynomial(coeffs:double...)` that takes a variable number of doubles to initialize the coeffs array, where the first argument corresponds to c_0. The three dots is known as *varargs* (variable number of arguments), which is a feature introduced in JDK 1.5. It accepts an array or a sequence of comma-separated arguments. The compiler automatically packs the comma-separated arguments in an array. The three dots can only be used for the last argument of the method.

- Another constructor that takes coefficients from a file (of the given `filename`), having this format:

```
Degree-n(int)
c₀(double)
c₁(double)
......
......
cₙ₋₁(double)
cₙ(double)
(end-of-file)
```

Hints:

```
public MyPolynomial(String filename) {
   Scanner in = null;
   try {
      in = new Scanner(new File(filename));  // open file
   } catch (FileNotFoundException e) {
      e.printStackTrace();
   }
   int degree = in.nextInt();      // read the degree
   coeffs = new double[degree+1];  // allocate the array
```

```
   for (int i=0; i<coeffs.length; ++i) {
      coeffs[i] = in.nextDouble();
   }
}
```

- A method `getDegree()` that returns the degree of this polynomial.
- A method `toString()` that returns "$c_n x^n+c_{n-1}x^{(n-1)}+...+c_1 x+c_0$".
- A method `evaluate(double x)` that evaluate the polynomial for the given `x`, by substituting the given `x` into the polynomial expression.
- Methods `add()` and `multiply()` that adds and multiplies this polynomial with the given `MyPolynomial` instance `another`, and returns a new `MyPolynomial` instance that contains the result.

Write the `MyPolynomial` class. Also write a test program (called `TestMyPolynomial`) to test all the methods defined in the class.

```
public class MyPolynomial {
   private double[] coeffs;
   public MyPolynomial(double... coeffs) {  // varargs
      this.coeffs = coeffs;                 // varargs is treated as array
   }
   ......
}

// Test program
// Can invoke with a variable number of arguments
MyPolynomial p1 = new MyPolynomial(1.1, 2.2, 3.3);
MyPolynomial p1 = new MyPolynomial(1.1, 2.2, 3.3, 4.4, 5.5);
// Can also invoke with an array
Double coeffs = {1.2, 3.4, 5.6, 7.8}
MyPolynomial p2 = new MyPolynomial(coeffs);
```