

# THE JAVA PROGRAMMING LANGUAGE

## POLYNOMIALS

University of Wrocław  
Institute of Computer Science

Paweł Rzechonek

### Exercise

A *polynomial* is a mathematical expression involving a sum of powers in one or more variables multiplied by coefficients. A polynomial in one variable (a univariate polynomial) with constant coefficients is given by:

$$P(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_2 x^2 + a_1 x + a_0 = \sum_{i=0}^n a_i x^i$$

The individual summands with the coefficients included are called monomials. The highest power in a univariate polynomial is called its order, or sometimes its degree (if  $n$  is the degree of a polynomial  $P(x)$  then  $a_n \neq 0$  for  $n > 0$ ).

Define a class `Polynomial`, which will represent a univariate polynomial. You should define the addition, subtraction and multiplication in this class.

```
public class Polynomial
{
    public final int deg; // degree
    private double[] c; // an array [0...n] with the coefficients

    // constructors
    public Polynomial () {/*...*/} // P(x) = 0 : c.length=0
    public Polynomial (int d) {/*...*/} // P(x) = x^d : c[deg]=1, c[deg-1]=...=c[0]=0
    public Polynomial (double a) {/*...*/} // P(x) = a : c[0]=a
    // P(x) = c[0] + c[1]*x + ... + c[deg]*x^deg : c[0]=a[0], ..., c[deg]=a[a.length-1]
    public Polynomial (int deg, double[] a) throws NullPointerException {/*...*/}

    // read the coefficient c[i]
    public double get (int i) throws IndexOutOfBoundsException {/*...*/}
    // set the coefficient c[i]=a
    public void set (int i, double a) throws IndexOutOfBoundsException, ArithmeticException {/*...*/}
    // evaluation of a polynomial P(x) - Horner scheme
    public double eval (double x) {/*...*/}

    public static Polynomial add (Polynomial first, Polynomial second) {/*...*/} // addition
    public static Polynomial sub (Polynomial first, Polynomial second) {/*...*/} // subtraction
}
```

```

public static Polynomial mult (Polynomial first, Polynomial second) {/*...*/} // multiplication
public static Polynomial mult (Polynomial poly, double c) {/*...*/} // multiply by a constant
public static Polynomial mult (double c, Polynomial poly) {/*...*/} // multiply by a constant

    public String toString () {/*...*/}
}

```

Evaluation of a polynomial consists of assigning a number to each variable and carrying out the indicated multiplications and additions. Define evaluation method `eval(double)` more efficiently using the Horner scheme:

$$((\dots (c_{deg}x + c_{deg-1})x + \dots + c_2)x + c_1)x + c_0$$

Finally write a program, which will test your `Polynomial` class. Get two polynomials  $P$  and  $Q$ , and calculate  $P + Q$ ,  $P - Q$ ,  $P * Q$ ,  $P * 2$ , and  $3 * Q$ , and write the results to the standard output. Calculate the values of the polynomials on the interval  $[-1, 1]$  with the step 0.1.

### Hint

A lot of interesting information about polynomials can be obtained on the website:

<http://en.wikipedia.org/wiki/Polynomial>