

Homework 2

September 15th 2019

Return your solutions (i.e., .cpp files) to tuomas.lunttila@uef.fi by September 29th.

Implement a class for computing with polynomials with (double precision) floating point coefficients. The class should overload addition, subtraction, multiplication, division, modulo (remainder), function call (evaluation), and stream insertion (output) operations (so you can get output formatted something like $2.1x^2+4.2x+1.1$). That is, you should overload `+`, `-`, `*`, `/`, `%`, `()`, and `<<`. Write also a member function that calculates the derivative of the polynomial, and basic accessor (getter/setter) functions.

In the "Homework Problems" folder on course Moodle pages you can see a class declaration for `Poly`, an example of a helper function, which chops off unneeded (almost) zeros from the coefficient vector (I stored the coefficients of the polynomial in a vector, with the first element storing the coefficient of the x^0 , the second is the coefficient of x^1 and so on), function prototype for overloading `<<`, and an example of a main program, which uses the polynomial class. Of course, you don't need to use any of these in your own implementation if you want to write something better. There's no need for anything especially fancy like advanced checking for errors (floating point rounding may cause problems in some computations, but you don't need to worry about those here). My own try at this was less than a hundred lines, so you don't need to write a very long solution.