Coding Problems 2

Due 6pm PDT Tuesday, Jul 23, 2019

Problem 1

- Write a Polynomial class that stores the parameters of a polynomial in a vector<double> a,

$$f(x) = \sum_{i=0}^{i<a.size()} a_i x^i$$

- With the following methods:
  - Polynomial(); //default constructor with a[0]=0.0
  - Polynomial(const vector<double>& b); // constructor 2
  - Polynomial(const Polynomial& p); // copy constructor
  - void set_parameters(const vector<double> b);
  - void set_parameter(int i, double b);
  - vector<double> get_parameters( ) const; // return vector of parameters
  - double get_parameter(int i) const; // return a[i]
  - double operator()(const double x) const; // evaluate f(x)
  - int get_order() const; // return order of polynomial
- Put the code in the files Polynomial.h and Polynomial.cpp

Problem 2

- Overload the << operator as a helper function in Polynomial.h and Polynomial.cpp with the following interface:

- ostream& operator<<(ostream& out, const Polynomial &p);

- That prints, for example, for a polynomial with vector<double> a = {1.0, -2.0, 3.0};

f(x) = 1.0 – 2.0 * x + 3.0 * x ^ 2

Problem 3

- Integrating a polynomial second version
- Use the polynomial class in Problem 1
- Include a helper function in Polynomial.h and Polynomial.cpp with interface:
- double integrate(const Polynomial& p, double xmin, double xmax);

Problem 4

- Roots of a polynomial
- Use the Polynomial class of Problem 1
- Include a helper function in Polynomial.h and Polynomial.cpp that finds the roots of a Polynomial with interface:
- vector< complex<double> > find_roots(const Polynomial & p);

Submission and testing

- A test main.cpp program will be sent to you by 6pm PDT on Mon Jul 22
- Hand in your Polynomial.h and Polynomial.cpp by 6pm PDT on Tue Jul 23, to:
  - bl.jamieson@uwinnipeg.ca,
  - jmgwalker@triumf.ca, and
  - MPavin@triumf.ca