수치 해석학 과제2 – Exercise1.2.10

1. About code

I made the Assignment2 class to solve each problems. Here, there are two functions that we want to find zeros. One is f1(x) = (x-1)\*(x^7+6x^6+3x^2-3) and the other is f2(x) = 2x^7-x^6-3.5x^4+2. For f1, there are two zeros near 0.757627 and -0.779945. For f2, there is one zero between -1 and 0. The Assignment2 class has three functions to find zeros of a given function.

First, **double SecantMethod(double(\*f)(double x), double x0, double x1, int maxloop, vector<double>& error, deque<double>& z\_seq)** is the Secant method for a function f. The inputs x0 and x1 are two initial points with x0<x1. The input maxloop is the maximum number of iterations so that the infinite loop can be avoided if the sequence is not convergence. The input error is a vector of convergence orders which are obtained from the z\_seq. The input z\_seq is a vector of last 10 errors (i.e. the difference between the sequence and the limit point that is the desired zero) to calculate convergence order. Here, I used the data structure ‘deque’ which is combination of queue and stack because this structure is easy to delete the front element and save element to the back of the z\_seq in order to save only last 10 errors.

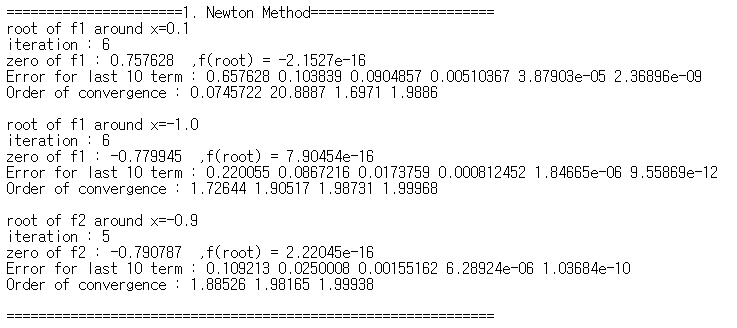
Second, **double NewtonMethod(double(\*f)(double x), double(\*df)(double x), double x0, int maxloop, vector<double>& error, deque<double>& z\_seq)** is the Newton method. Thus, this function has two important functions: one is a function to find zeros and the other is its derivative. The input x0 is the initial point. The other inputs are same as the Secant method.

Last, **double SteffensenMethod(double(\*f)(double x), double x0, int maxloop, vector<double>& error, deque<double>& z\_seq)** is the Steffensen’s method for a function f. Other inputs are same as the Newton method.

The stopping criteria of these functions is that either the function value is very close to zero or the difference between the previous point and the next point is very close to zero.

1. Newton’s method

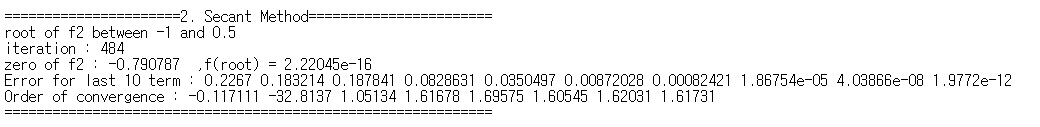
For f1, I choose initial points 0.1 and -1 to find two zeros respectively. For f2, I choose the initial point -0.9. The result is as follow:



As the result, f1 has 0.75768 and -0.779945 as its roots and f2 has -0.790787 as a root. Although the number of iterations to find zero is somewhat small in each case, the convergence order goes to 2 for all three cases.

1. Secant Method

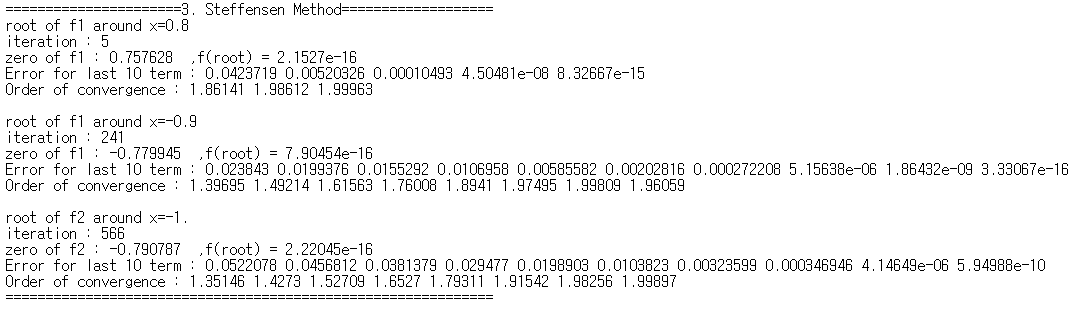
For f2, I choose initial points -1 and 0 to find a zero between -1 and 0.5. The result is as follow:



As the result, f2 has -0.790787 as a root, which is the same result in the Newton’s method. The last convergence order is 1.61731, which is close to the theoretical order 1.618.

1. Steffensen’s method

For f1, I choose initial points 0.8 and -0.9 to find two roots, respectively and for f2, I choose an initial point -1. The result is as follow:



As the result, f1 has 0.75768 and -0.779945 as its roots and f2 has -0.790787 as a root, which is the same result in the Newton’s method. In any case, the convergence order goes to 2.