The Report for Programming Assignments in Chapter One

Wenchong Huang

Date: October 5, 2022

1 How to Test

Enter the folder `Programming-Chapter1/src` with terminal, `make` here, you will see some executable files whose names are corresponding assignments. Run them directly and you will see the results.

2 Manual

This package is for solving equations $f(x) = 0$.

You should include the header `EquationSolver.h`, then define your function object as a derived class of `Function`. The deriviate function is an optional. If you didn't define the deriviate function but still use Newton's Method, the solver will use difference quotient to replace your derivate. Here is an example where $f(x) = x^{-1} - \tan x$.

```
class F : public Function{
public:
    double operator () (const double &x) const{
        return 1.0/x-tan(x);
    }
    double diff (const double &x) const{
        return -1.0/(x*x) - 1.0/(cos(x)*cos(x));
    }
} f;
```

You could use Bisection Method to solve the equation $f(x) = 0$, as following, where the `a`, `b`, `delta` , `eps` and `M` are $a, b, \delta, \varepsilon$ and M in the Algorithm 1.1 of the textbook.

```
  BisectionSolver sol(f, a, b, delta, eps, M);
  double ans = sol.solve();
```

You may also try Newton's Method, as following, where the `x0` , `eps` and `M` are x_0, ε and M in the Algorithm 1.14 of the textbook.

```
  NewtonSolver sol(f, x0, eps, M);
```

```
double ans = sol.solve();
```

You may also try Secant Method, as following (Don't ask why the name is `SecandMethod`, that's a secre**d**.), where the `x0`, `x1`, `delta` , `eps` and `M` are $x_0, x_1, \delta, \varepsilon$ and M in the Algorithm 1.19 of the textbook.

```
SecandSolver sol(f, x0, x1, delta, eps, M);
double ans = sol.solve();
```

Additional, if you don't want the solver's outputs to disturb you, you can add

```
#define SILENCE
```

before including this package.

3 Results

3.1 Assignment B

The results see the following table, where the δ and ε are all chose to be the machine precision.

$f(x)$	$[a,b]$	x^*	$f(x^*)$	k	$\lvert b_k - a_k \rvert$
$x^{-1} - \tan x$	$[0, \frac{\pi}{2}]$	0.860334	0	53	3.33067×10^{-16}
$x^{-1} - 2^x$	$[0, 1]$	0.641186	0	52	4.44089×10^{-16}
$2^{-x} + e^x + 2\cos x - 6$	$[1, 3]$	1.82938	0	53	4.44089×10^{-16}
$\frac{x^3+4x^2+3x+5}{2x^3-9x^2+18x-2}$	$[0, 4]$	0.116378	6.08536×10^{15}	56	1.11022×10^{-16}

The Bisection Method performs well at the first three functions, but fails at the last. That's because the precondition $\operatorname{sgn}(f(a)) \neq \operatorname{sgn}(f(b))$ was not satisfied. Infact the last function has no root in interval $[0, 4]$. See the following image.

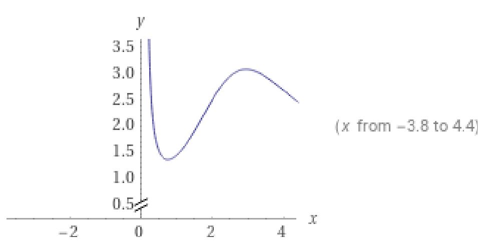

Figure 1: The image of the last function.

3.2 Assignment C

The results see the following table.

The Newton's Method converges fast as expected.

$f(x)$	x_0	x^*	$f(x^*)$	k
$x - \tan x$	4.5	4.49341	-8.88178×10^{-16}	4
	7.7	7.72525	2.30926×10^{-14}	5

3.3 Assignment D

The results see the following table.

$f(x)$	x_0	x_1	x^*	$f(x^*)$	k	$\lvert x_k - x_{k-1}\rvert$
$sin(x/2) - 1$	0	$\frac{\pi}{2}$	3.14159	-1.11022×10^{-16}	38	1.50313×10^{-8}
$e^x - \tan x$	1	1.4	1.30633	-4.44089×10^{-16}	17	0
$x^3 - 12x^2 + 3x + 1$	0	-0.5	-0.188685	0	9	0

We could try other initial values and get different results. See the following table.

$f(x)$	x_0	x_1	x^*	$f(x^*)$	k	$\lvert x_k - x_{k-1}\rvert$
$sin(x/2) - 1$	5.2π	5.5π	15.708	0	37	3.65908×10^{-8}
$e^x - \tan x$	-1.4	-1.5	-59.6903	1.22588×10^{-15}	15	0
$x^3 - 12x^2 + 3x + 1$	0	0.5	0.451543	2.22045×10^{-16}	9	1.11022×10^{-16}
	10	11	11.7371	1.63425×10^{-13}	10	0

The results are easy to understand. The function $sin(x/2) - 1$ vanishes at every point $(4k+1)\pi$ $(k \in \mathbb{Z})$. The function $e^x - \tan x$ has a root at each interval $[k\pi - \frac{\pi}{2}, k\pi + \frac{\pi}{2}]$ $(k \in \mathbb{Z})$. And the polynimial function $x^3 - 12x^2 + 3x + 1$ has three different real roots.

3.4 Assignment E

First define function $f(h) = L\left[0.5\pi r^2 - r^2 \arcsin\frac{h}{r} - h(r^2 - h^2)^{\frac{1}{2}}\right] - V$.

To make Newton's Method more efficient, we calculated the derivate by hand.

$$f'(h) = L\left[-\frac{r}{\sqrt{1 - \left(\frac{h}{r}\right)^2}} - \sqrt{r^2 - h^2} + \frac{2h^2}{\sqrt{r^2 - h^2}}\right]$$

The results see the following table.

	$[a, b]$	x_0	x_1	x^*	$f(x^*)$	k
Bisection Method	$[0, 1]$	*	*	0.167969	-0.0355476	8
Newton's Method	*	0	*	0.166177	-0.000215002	2
Secant Method	*	0	0.5	0.16623	-0.00126434	3

3.5 Assignment F

The function $f(x)$ defines as following.

$$f(x) = A \sin x \cos x + B \sin^2 x - C \cos x - E \sin x$$

Also, to make Newton's Method more efficient, we calculated the deriviate by hand.

$$f'(x) = A(-\sin^2 x + \cos^2 x) + 2B \cos x \sin x + C \sin x - E \cos x$$

The results see the following table.

		x_0	x_1	x^*	$f(x^*)$	k
(a)	Newton's Method	$33°$	*	$0.575473\ (32.9722°)$	0	3
(b)	Newton's Method	$33°$	*	$0.578907\ (33.1689°)$	0	5
(c)	Secant Method	$30°$	$33°$	$0.575473\ (32.9722°)$	0	5

We tried another initial value at problem (c), the result sees following.

		x_0	x_1	x^*	$f(x^*)$	k
(c)	Secant Method	$0°$	$5°$	$-0.200713\ (-11.5°)$	0	9
		$100°$	$120°$	$2.56612\ (147.028°)$	-3.55271×10^{-15}	9
		$180°$	$170°$	$2.94088\ (168.5°)$	-1.77636×10^{-15}	8

Actually, the function is periodic function with period 2π, and it has four different roots at each period. The image sees following.

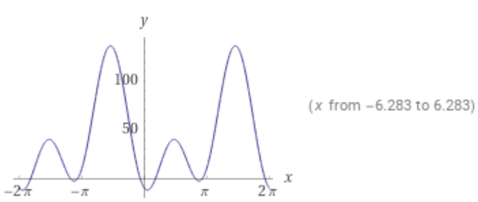

Figure 2: The image of the function in problem (c).

4 Summary

This template is designed by ElegantLaTeX Program.

Many appreciations for your carefully reading. If you found any mistakes, please contact me directly. Have fun with your loving one (boyfriend, girlfriend or coding)!