Math 128A Programming Assignment #1

Hannah Warner (SID: 3034613336)

1) findzero

```
function p = findzero(f, a, b, tol)

w = 1;

% print header
fprintf('      a              b              p            f(p)     \n');
fprintf('---------------------------------------------------------\n');

% iterates at most 100 times
for n = 1:100

    % compute p and function f at p
    p = a + (w * f(a) * (a - b)) / (f(b) - w * f(a));
    f_p = f(p);

    % output row of results
    fprintf('%12.8f  %12.8f  %12.8f  %12.8f\n', a, b, p, f(p));

    % set variables
    if f_p * f(b) > 0
        w = 1 / 2;
    else
        w = 1;
        a = b;
    end
    b = p;

    % check terminating conditions
    if abs(b - a) < tol, break; end
    if abs(f_p) < tol, break; end

end

end
```

2) solve f(x) = cos(x) - x with a = 0, b = 1, and tol = 10^{-10}

```
>> f = @(x) cos(x) - x
>> findzero(f, 0, 1, 10e-10)

      a              b              p            f(p)
---------------------------------------------------------
  0.00000000    1.00000000    0.68507336    0.08929928
  1.00000000    0.68507336    0.73629900    0.00466004
  1.00000000    0.73629900    0.74153913   -0.00410926
  0.73629900    0.74153913    0.73908362    0.00000253
  0.74153913    0.73908362    0.73908513    0.00000000
  0.74153913    0.73908513    0.73908513   -0.00000000
  0.73908513    0.73908513    0.73908513    0.00000000
```

```
ans =

    0.7391
```

It appears as if f(x) has at least a quadratic order of convergence. We see that the function only took a few iterations before evaluating a zero with difference less than tol, so we know the order of convergence must be high.

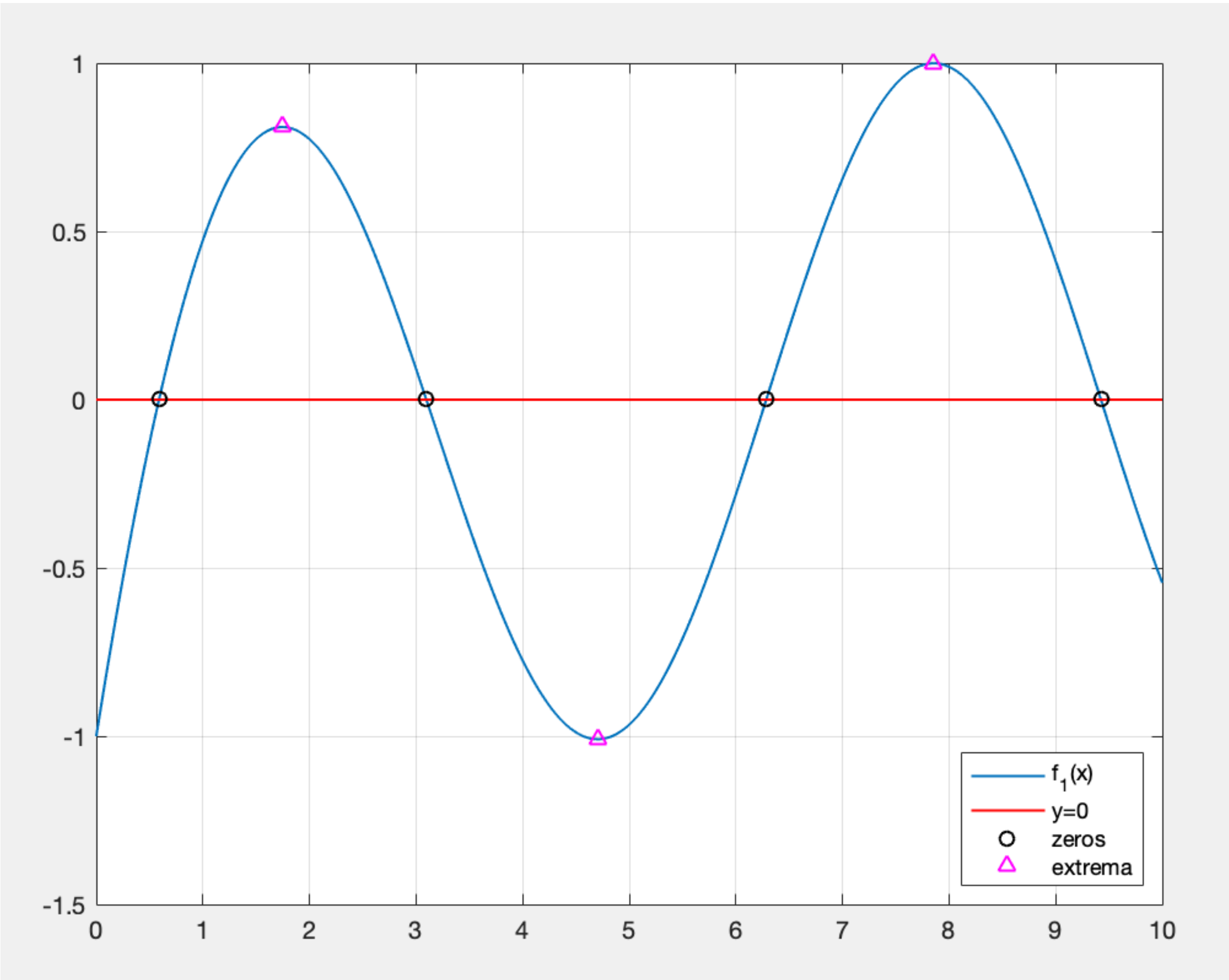
3) findmanyzeros

```
function p = findmanyzeros(f, a, b, n, tol)

% initialize output vector
p = [];

% calculate interval width and x vector
interval_width = (b - a) / n;
xs = a:interval_width:b;

% iterate n times - compute zero using findzero if condition applies
for i = 2:n + 1
    if f(xs(i)) * f(xs(i - 1)) < 0
        p_i = findzero(f, xs(i), xs(i - 1), tol);
        % concatenate new computed zero to end of output vector
        p(end + 1) = p_i;
    end
end

end
```

4) solve and plot: $f_1(x) = \sin(x) - e^{-x}$
$f_2(x) = \sin(x^2) / (10 + x^2) - e^{-x/10} / 50$

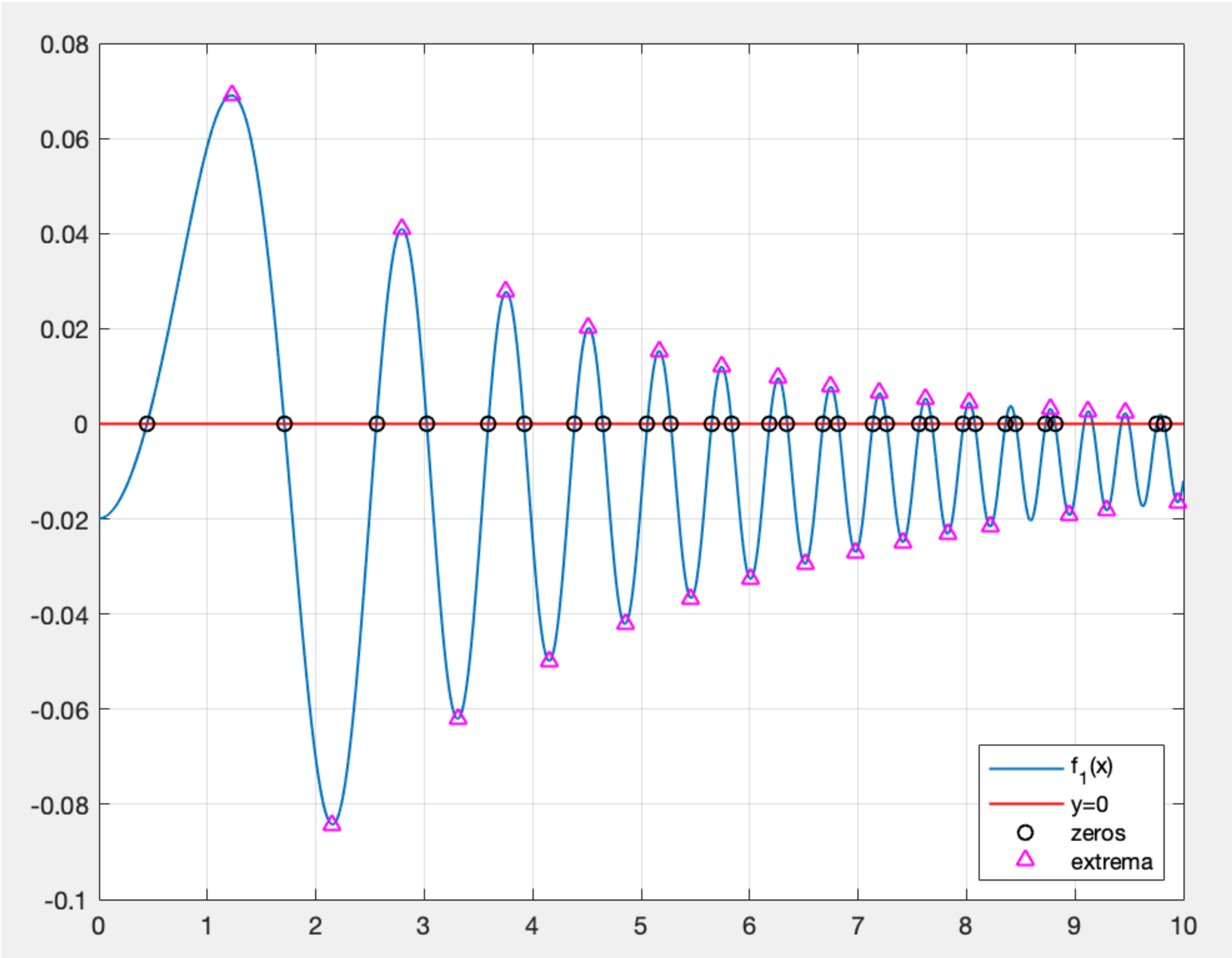
```
function findmanyzeros_plot(f, f_div, a, b, n, tol)

clf

% plot function f
xx = linspace(0, 10, 1000);
plot(xx, f(xx), 'linewidth', 1, 'displayname', 'f_1(x)');
hold on

% determine zeros of f
p = findmanyzeros(f, a, b, n, tol);

% plot line y=0 and zeros of f
line([0, 10], [0, 0], 'linewidth', 1, 'color', 'r', 'displayname', 'y=0');
plot(p, f(p), 'ko', 'linewidth', 1, 'displayname', 'zeros');

% calculate zeros of derivative and plot extrema of f
p_derivative = findmanyzeros(f_div, a, b, n, tol);
plot(p_derivative, f(p_derivative), 'm^', 'linewidth', 1, 'displayname',
'extrema');
```

```
legend('location', 'southeast')
grid on

end

>> f1 = @(x) sin(x) - exp(-x)
>> f1_div = @(x) cos(x) + exp(-x)
>> findmanyzeros_plot(f1, f1_div, 0, 10, 50, 10e-10)
```



We see from the graph that $f_1(x)$ has four zeros and three extrema on the interval [0, 10]. The zeros of the function are at x-values 0.5885, 3.0964, 6.2850, and 9.4247.

```
>> f2 = @(x) (sin(x.*x) ./ (10 + (x.*x))) - (exp(-x/10) / 50)
>> f2_div = @(x) ((-2*x.*sin(x.*x)) ./ ((x.*x + 10).*(x.*x + 10))) +
((2*x.*cos(x.*x)) ./ (x.*x + 10)) + (exp(-x/10) / 500)
>> findmanyzeros_plot(f2, f2_div, 0, 10, 50, 10e-10)
```



We see from the graph that $f_2(x)$ appears to begin to converge on the interval [0, 10]. The function has many zeros and extrema, however due to the interval size n we input to the function, a few in the range [8, 10] do not appear in the plotted output. If we were to apply a larger n value, such as 200 rather than 50, these zeros and extrema would be calculated and subsequently appear.