MAT 4010: Homework 5

Instructions

- Please write clearly and show all of your work. Answers without justification may be worth zero.
- When justifying your conclusions using a theorem, give the theorem name/number.
- Submit your homework to the appropriate assignment on Gradescope before the deadline.
- If you complete your homework on paper, you are encouraged to use a scanner app to convert photos of your work to a PDF before submitting to Gradescope.
- For programming questions, submit your source code (.m files) and any plots generated (.png files) to the appropriate "Programming" assignment on Gradescope before the deadline. Your .m files must follow the naming convention Hxx_Qyy.m where xx is a two-digit homework number and yy is a two-digit question number (e.g., H05_Q02.m).

Problem Set

1. Let $f(x) = x^2 - 5$.

   a) Find the corresponding fixed point function $g(x)$ for Newton's Method.

   b) Starting with the initial guess $p_1 = 2.5$, run Newton's Method by hand to complete the following table. Keep as many digits of precision as you calculator gives (hopefully at least 8) in intermediate calculations and from one iteration to the next, but round to 7 significant figures (which will mean 6 digits after the decimal) when filling in the table to turn in.

n	p_n
1	2.5
2	
3	
4	

2. In this problem we will investigate using Newton's Method to approximate a root of $f_1(x) = x^2 - 5$ and also of $f_2(x) = (x^2 - 5)^2$. Note: Since Newton's Method is a type of fixed point iteration, you may wish to modify code you have previously written.

   Write a program (using MATLAB/Octave) that implements Newton's Method as discussed in class. Your program should obtain from the user the function f for which a root is to be estimated and its derivative f', both as anonymous functions, as well as an initial p_1 and values for ϵ and the maximum n. You code should assume that $\alpha = 2$. Each iteration $n \geq 2$, your program should print the current n, p_n, and $\hat{e}_{n-1}$. For values of n for which $\hat{e}_n$ can not be computed, use 10ϵ. Note that in MATLAB/Octave if you have anonymous functions `f` and `fprime` you can do

```
g = @(x) x - f(x) ./ fprime(x);
```

Your program should also produce three plots. The first plot should show p_n versus n, including appropriate axes labels. The second plot should show $\hat{e}_n$ versus $\hat{e}_{n-1}$. When labelling the axes, you can use

```
xlabel( '$\hat{e}_{n-1}$', 'interpreter', 'latex' );
```

For a title you can use (assuming your annonymous function variable is `f`)

```
title( sprintf( 'Newton''s Method for f(x) = %s', func2str(f) ) );
```

The third plot should show $\hat{\alpha}_n$ versus n where (recalling from Homework 1) we have

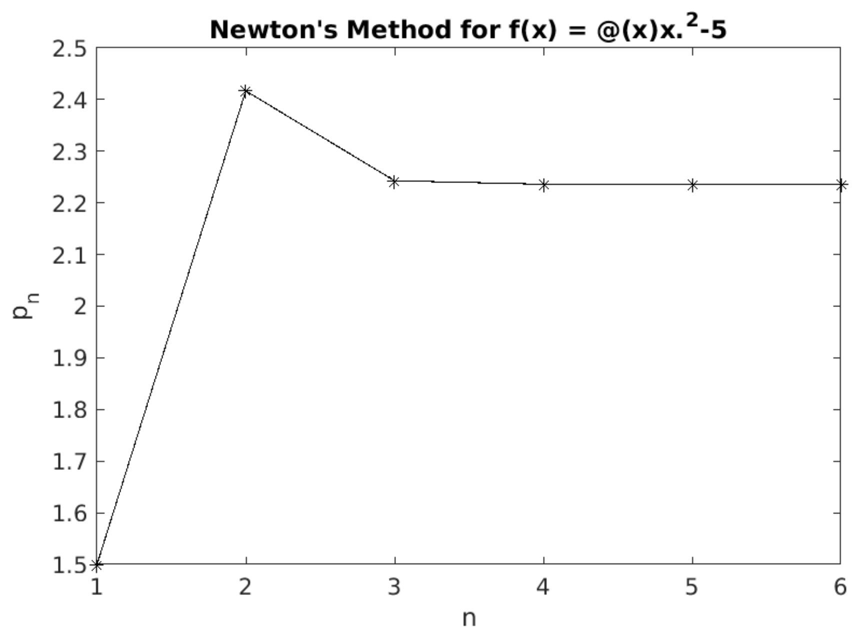
$$\alpha \approx \frac{\ln e_n - \ln e_{n-1}}{\ln e_{n-1} - \ln e_{n-2}} \approx \frac{\ln \hat{e}_n - \ln \hat{e}_{n-1}}{\ln \hat{e}_{n-1} - \ln \hat{e}_{n-2}} \equiv \hat{\alpha}_n$$

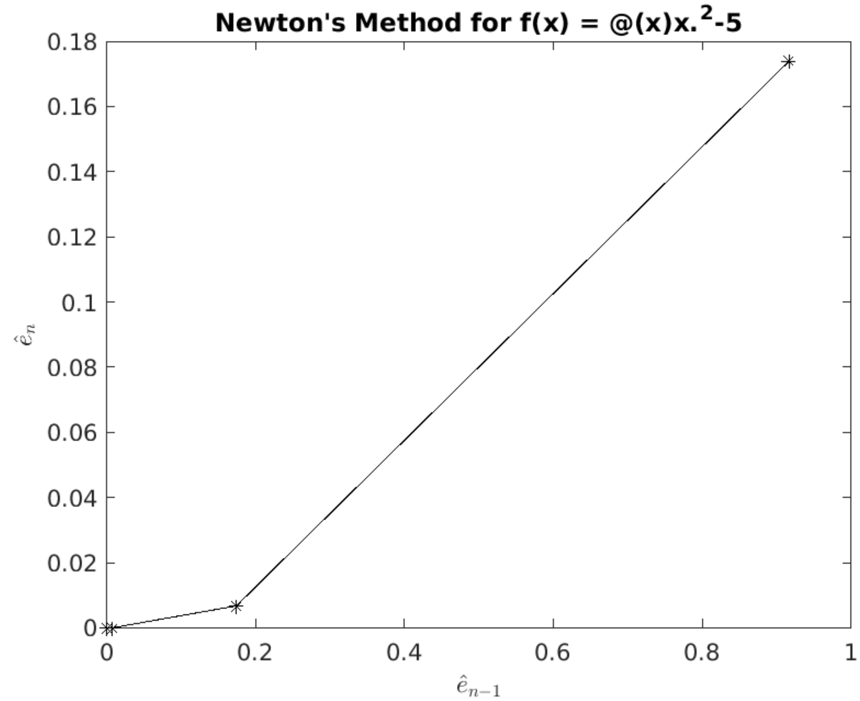
Note: Comment out any plotting-related code before submitting to Gradescope.

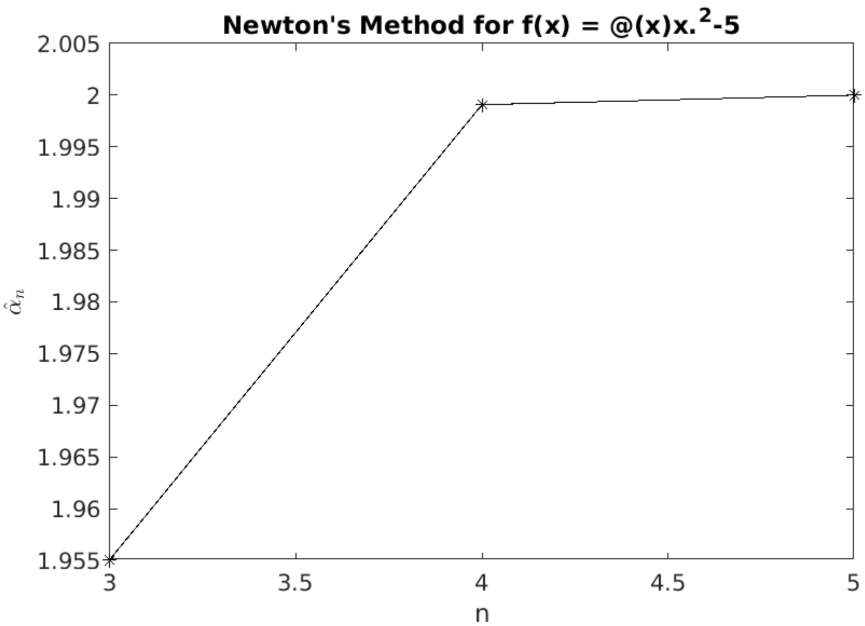
Sample output:

```
Enter a value for p1: 1.5
Enter an anonymous function f: @(x) x.^2 - 5
Enter an anonymous function f': @(x) 2 * x
Enter a value for epsilon: 1e-6
Enter a maximum value for n: 10
n =  2: p(2) =  2.41666667, ehat(1) = 9.16666667e-01
n =  3: p(3) =  2.24281609, ehat(2) = 1.73850575e-01
n =  4: p(4) =  2.23607813, ehat(3) = 6.73796270e-03
n =  5: p(5) =  2.23606798, ehat(4) = 1.01517341e-05
n =  6: p(6) =  2.23606798, ehat(5) = 2.30442332e-11
```

Sample plots:




Newton's Method for f(x) = @(x)x.^2-5
0.18
0.16
0.14
0.12
0.1
0.08
0.06
0.04
0.02
0
0.2
0.4
0.6
0.8
1
$\hat{e}_n$
$\hat{e}_{n-1}$


Newton's Method for f(x) = @(x)x.^2-5
2.005
2
1.995
1.99
1.985
1.98
1.975
1.97
1.965
1.96
1.955
3
3.5
4
4.5
5
$\hat{\alpha}_n$
n

Sample Output:

```
Enter a value for p1: 1000
Enter an anonymous function f: @(x) x.^2 - 5
Enter an anonymous function f': @(x) 2 * x
Enter a value for epsilon: 1e-6
Enter a maximum value for n: 10
n =  2: p(2) = 500.00250000, ehat(1) = 4.99997500e+02
n =  3: p(3) = 250.00624998, ehat(2) = 2.49996250e+02
n =  4: p(4) = 125.01312474, ehat(3) = 1.24993125e+02
n =  5: p(5) = 62.52656027, ehat(4) = 6.24865645e+01
n =  6: p(6) = 31.30326314, ehat(5) = 3.12232971e+01
n =  7: p(7) = 15.73149545, ehat(6) = 1.55717677e+01
n =  8: p(8) =  8.02466459, ehat(7) = 7.70683086e+00
n =  9: p(9) =  4.32387180, ehat(8) = 3.70079280e+00
n = 10: p(10) =  2.74012140, ehat(9) = 1.58375039e+00
Error: Newton's Method did not converge
```

Sumbit: Your .m file and the .png files for inputs of

```
p1 = 1.5
f = @(x) ( x.^2 - 5 ).^2
fprime = @(x) 4 * ( x.^2 - 5 ) .* x
epsilon = 1e-6
n = 20
```