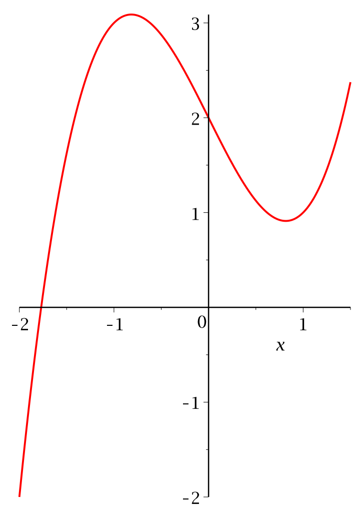
Math 1310 Lab 9. (Sec 4.7 - Sec 4.8)

Name/Unid: ______________________________ Lab section: ____

1. **(Newton's method: Oops.)**

   The following figure is the graph for $f(x) = x^3 - 2x + 2$ for $-2 \leq x \leq 2$. There's a root for $f(x) = 0$ between -2 and -1.

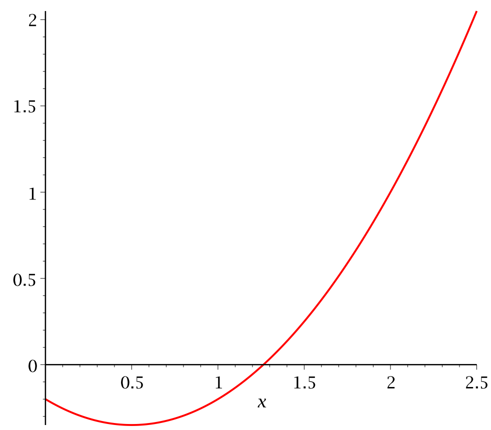
   

   Newton's method is one approach to find the root. We proceed with some initial value x_0, and get $x_1, x_2, \cdots$ by $x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$, for $n \geq 0$.

   Question 1. Let $x_0 = -1.7$. Use a calculator to find x_1, x_2, x_3 (round to three decimal digits each time you iterate). Does it seem to converge or not? **(2 pts)**

   Question 2. Let $x_0 = 0$. Use Newton's method to find x_1, x_2, x_3, x_4, x_5. Draw two tangent lines: the one passing through $(0, 2)$, and the one through $(1, 1)$ respectively, on the figure above. Briefly explain why the method doesn't converge. **(3 pts)**

Question 3. Here is the figure for another function.



Now you have 4 choices as the initial value for Newton's method: (a)0.7 (b)1 (c)1.5 (d)2. What choices will make Newton's method work? Compute at least five iterations for each initial value. **(8 pts)**

Solution: A1. $x_1 = -1.773$, $x_2 = -1.769$, $x_3 = -1.769$.
A2. $x_1 = x_3 = x_5 = 1$, and $x_2 = x_4 = 0$. There's a cycle of period 2 for these values.
A3. They all work.

2. **(What happened to the anti-derivatives?)**

Question 1. If you differentiate $\ln x$ with respect to x, you will get $\frac{1}{x}$. If you differentiate $\ln(2x)$ with respect to x, by the chain rule, you'll get $\frac{1}{2x} \cdot 2 = \frac{1}{x}$. This holds for $\ln(3x), \ln(4x), \cdots$ as well. This means, $\ln x, \ln(2x), \ln(3x), \cdots$ are all anti-derivatives for $\frac{1}{x}$ (consider $x > 0$). Theorem 1 in section 4.8 tells that if $F(x)$ is an antiderivative of $f(x)$, then any other antiderivative of $f(x)$ has the form $F(x) + C$, where C is some number. Explain why the ones listed above are all antiderivatives of $\frac{1}{x}$. Then find two different forms for the general antiderivative of $\frac{1}{x}$ (assume $x > 0$) and show how to go from one to the other and the other way around. **(3 pts)**

Solution: $\ln 2x = \ln x + \ln 2$. Because $\ln 2x$ and $\ln x$ only differ by a constant, it is possible that they are both antiderivatives for a function. Then we can write the general antiderivative of $\frac{1}{x}$ (for $x > 0$) as $\ln(x) + C$, where C is any number or as $\ln(Dx)$, where D is any positive number. Since the logarithm of the product is the sum of the logarithms, we have that $D = e^C$ and $C = \ln(D)$.

3. **(The parabolic motion)**

For future space exploration, NASA plans to build a base on the moon's surface. To get started, some material has to be thrown on the surface of the moon. A space shuttle is in orbit around the moon at a height of $3000m$ with a constant speed of $100m/s$. The gravity of the moon is $1.62m/s^2$. If some material is thrown from the space shuttle, the motion is described by $x(t) = 100m/st$ and $y(t) = at^2 + bt + c$.

Question 1. We know that the gravity on the moon is $1.62m/s^2$, that the initial velocity of the material in the vertical direction is $0m/s$ and we know that the space shuttle is in orbit at a height of $3000m$. Use this data to determine a, b and c. Remember to include the units of measure! **(3 pts)**

Question 2. Determine how far the material travels in the x direction before hitting the ground. **(3 pts)**

Question 3. If we know that the material has velocity $v_x(t)$ in the x direction and $v_y(t)$ in the y direction, the magnitude of the overall velocity is given by $v(t) = \sqrt{v_x^2(t) + v_y^2(t)}$. Some material is very fragile, so a special parachute has been projected. The parachute needs an overall velocity of at least $120m/s$ to be opened and once it opens, the magnitude of the velocity remains stable (i.e. if we open the parachute at time t_0, then for $t \geq t_0$ we will have $v(t) = v(t_0)$). To be sure the material does not break, the overall velocity cannot exceed $130m/s$ at the impact. What is the time interval during which the parachute can be opened so that the material does not break? **(3 pts)**

Solution:

(a) We are give the position $y(t) = at^2 + bt + c$ and we know that the velocity is its derivative $y'(t) = 2at + b$ and the acceleration is $y''(t) = 2a$. We have $y(0) = 3000m$ and $y'(0) = 0m/s$ and $y''(t) = -1.62m/s^2$. This implies $c = 3000m$, $b = 0m/s$ and $a = -0.81m/s^2$.

(b) We first have to find how long it takes to hit the ground. Hence, we set $y(t) = 0m$, which leads to the equation $-0.81m/s^2t^2 + 3000m = 0m$, which has solution (we look for t positive) $60.86s$. If we plug it into $x(t)$ we get $6086m$.

(c) We have $v_y(t) = -1.62m/s^2t$ and $v_x(t) = 100m/s$. Thus we have $v(t) = \sqrt{2.6244t^2 + 10000}m/s$. Thus we first set $v(t) = 120m/s$, which gives (we look for $t > 0$) $t = 40.95s$. Analogously, we do this for $v(t) = 130m/s$ and we get $t = 51.28s$. Thus, we get that the suitable time interval is $[40.95s, 51.28s]$.

4. **(Some MATLAB)**

Here are a few examples of MATLAB codes

```
% script calculating and plotting a(n)=(1+1/n)^n, n>0
% N.B. e=exp(1)= limit as n goes to infinity of a(n)
clear % clear all variables already defined
close all % close all the windows with graphs
n=[1:1000]; % we will evaluate a(n) for n from 1 to 1000, so we take
% an array of length 1000
a=(1+1./n).^n; % in another array of length 1000 we store a(n)
% in position n
plot(a,'o') %we plot it
hold on %we say we want to plot again on the same graph
plot([1 1000],[exp(1) exp(1)],'r-') % plot the line y=e
```

```
%example of cycle if
clear;
N=input('Type an integer number N\n'); %\n is needed to display N in
% a new line
M=input('Type another integer M you want to compare N to\n');
if N==M %we need double equal sign when we compare
    answer='N equals M';
elseif N>M
    answer='N is bigger than M';
elseif N<M
    answer='N is smaller than M';
end
answer %if we do not put a semicolumn at the end of a line MATLAB dispays it
```

```
%example of cycle while
clear;
N=input('Type an integer number N\n'); %\n is needed to display N in
% a new line
M=input('Type how many times you want to add 5 to N\n');
a=1; %we use a to keep track of how many times we have added 5
while a<=M
    N=N+5; %we add 5 to N and we keep using the same letter for the
    %new number we get
    a=a+1; %we add 1 to a each time we add 5
end
N
```

```
%Example of taking a derivative with an algorithm
clear;
f=@(x) x^3; %we need @(x) to tell MATLAB x is the variable
g=@(x) 3*x^2; %* is times
N=input('Type a number where you want to evaluate the derivative of x^3\n')
disp('The actual values is:')
g(N)
%We try to approximate f'(x) with (f(x+h)-f(x-h))/2h
h=0.1; %we first try with this h
D=(f(N+h)-f(N-h))/(2*h);
%We try to stay close at least 0.05
while abs(D-g(N))>0.05 %abs is the absolute value, a function in MATLAB
    h=h/2;
    D=(f(N+h)-f(N-h))/(2*h);
end
disp('The approximation we got is:')
D
%Since in general we do not know f', we cannot compare the approximation
%with the real value. For this reason, we decide to stop when
```

```
two
%successive approximations are very close
k=0.5;
E=(f(N+k)-f(N-k))/(2*k);
F=(f(N+(k/2))-f(N-(k/2)))/(k);
%We decide to stop when two successive approximation are not
%further than 0.1
while abs(E-F)>0.1
    k=k/2;
    E=F;
    F=(f(N+(k/2))-f(N-(k/2)))/(k);
end
disp('The approximation we got this way is:')
E
```

This script contains the Newton's method applied to the example in question 1.

```
%Now we will do with MATLAB what we did by hand
clear all;
close all;
f=@(x) x^3-2*x+2;
g=@(x) 3*x^2-2;
fplot(@(x) x^3-2*x+2,[-3,3])
%Since in this case we know the actual derivative, we will use
    it
N=-2; %we start Newton's method from here
%We decide to stop Newton's method after 10 iterations
c=1; %we use it to count the iterations
while c<=10
    N=N-f(N)/g(N);
    c=c+1;
end
disp('The approximation of the zero we got is:')
N
disp('If we plug it into the function we get:')
f(N)
```

Question 1. Write a MATLAB code for the example in question 1. To achieve this, modify the given script changing the arresting criterion in the while cycle. More precisely, use a while cycle that stops when $|f(x_n)| < 0.01$. Write the new code here. Furthermore, run the program on a computer and print the result. **(5 pts)**