1. a)

function error = error\_using\_forward(h)

x = 2;

exact\_derivative = exp(x);

forward\_computed = (exp(x + h) - exp(x))/h;

error = log10((forward\_computed - exact\_derivative)/exact\_derivative);

end

function error = error\_using\_central(h)

x = 2;

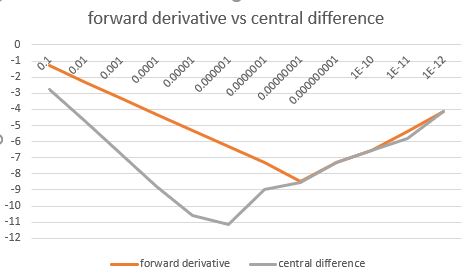
exact\_derivative = exp(x);

central\_computed = (exp(x + h) - exp(x - h))/(2 \* h);

error = log10((central\_computed - exact\_derivative)/exact\_derivative);

end

b)



c) I can find the gradient of the forward derivative as it is two straight lines. Abs(ax) – 8.5

I can find the gradient for the central difference until 10^-5 and then the line takes an ugly turn.

Both lines show that the formulas become more correct until 10^-6 and 10^-8 respectively. After this point both lines show a numerical error as their numbers become to big.

1. IN MATLAB:

syms a b c h;

function1 = a + b + c;

function2 = h\*(b + 2\*c) - 1;

function3 = b + 4\*c;

sol = solve([function1, function2, function3], [a,b,c]);

aSol = sol.a

bSol = sol.b

cSol = sol.c

BY HAND:

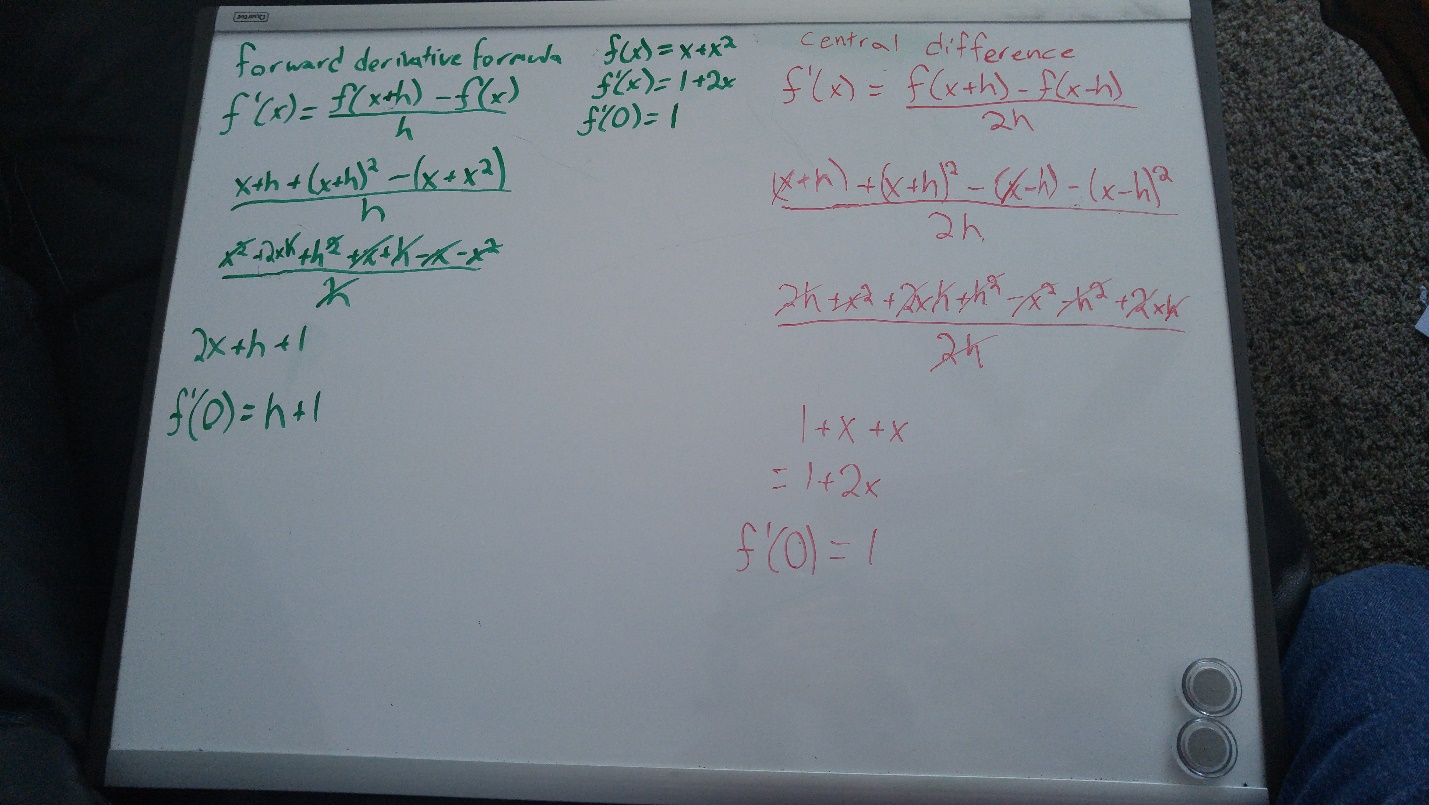
1 1 1 | 0 1 0 -3 | 0 1 0 -3 | 0 1 0 0 | -3/2h

0 h 2h | 1 0 1 4 | 0 0 1 4 | 0 0 1 0 | 4/2h

0 1 4 | 0 0 0 -2h | 1 0 0 1 | -1/2h 0 0 1 | -1/2h

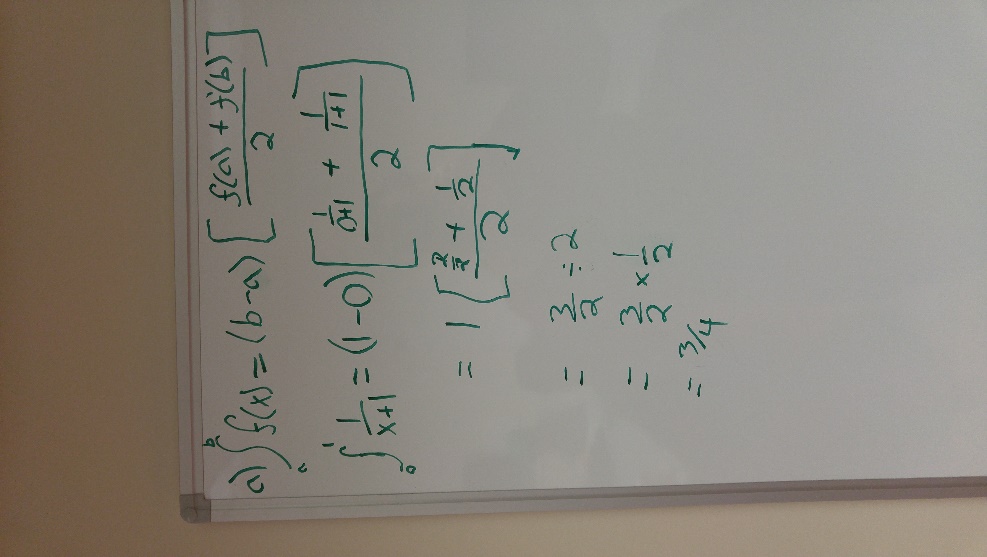
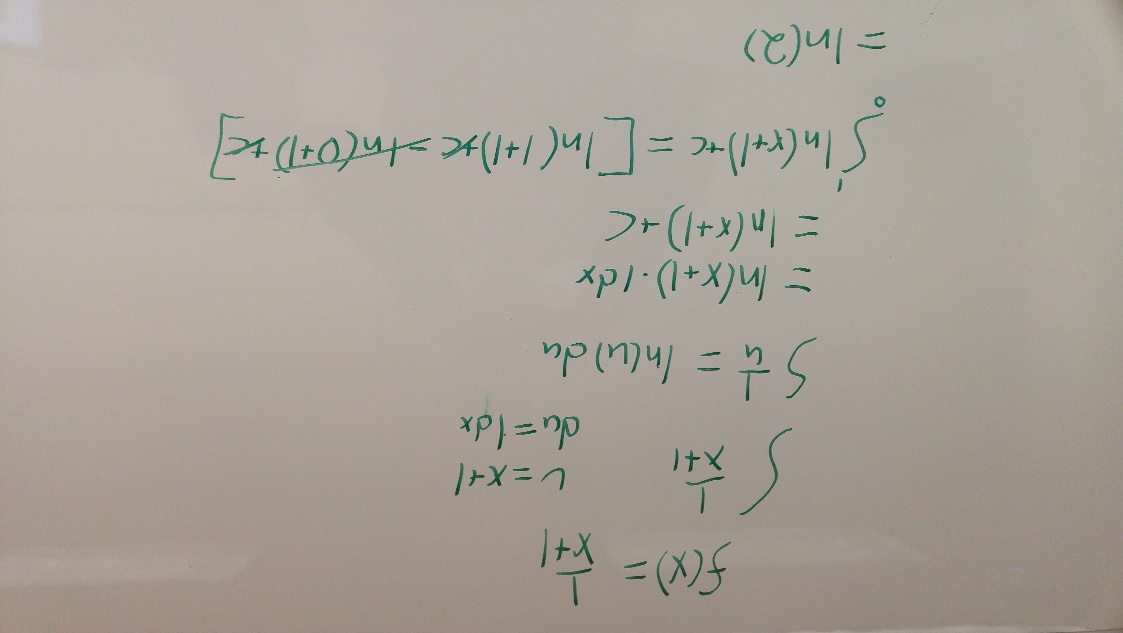
a = -3/2h b = 4/2h c = -1/2h

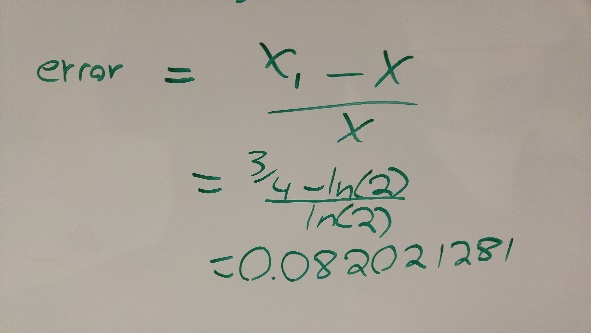


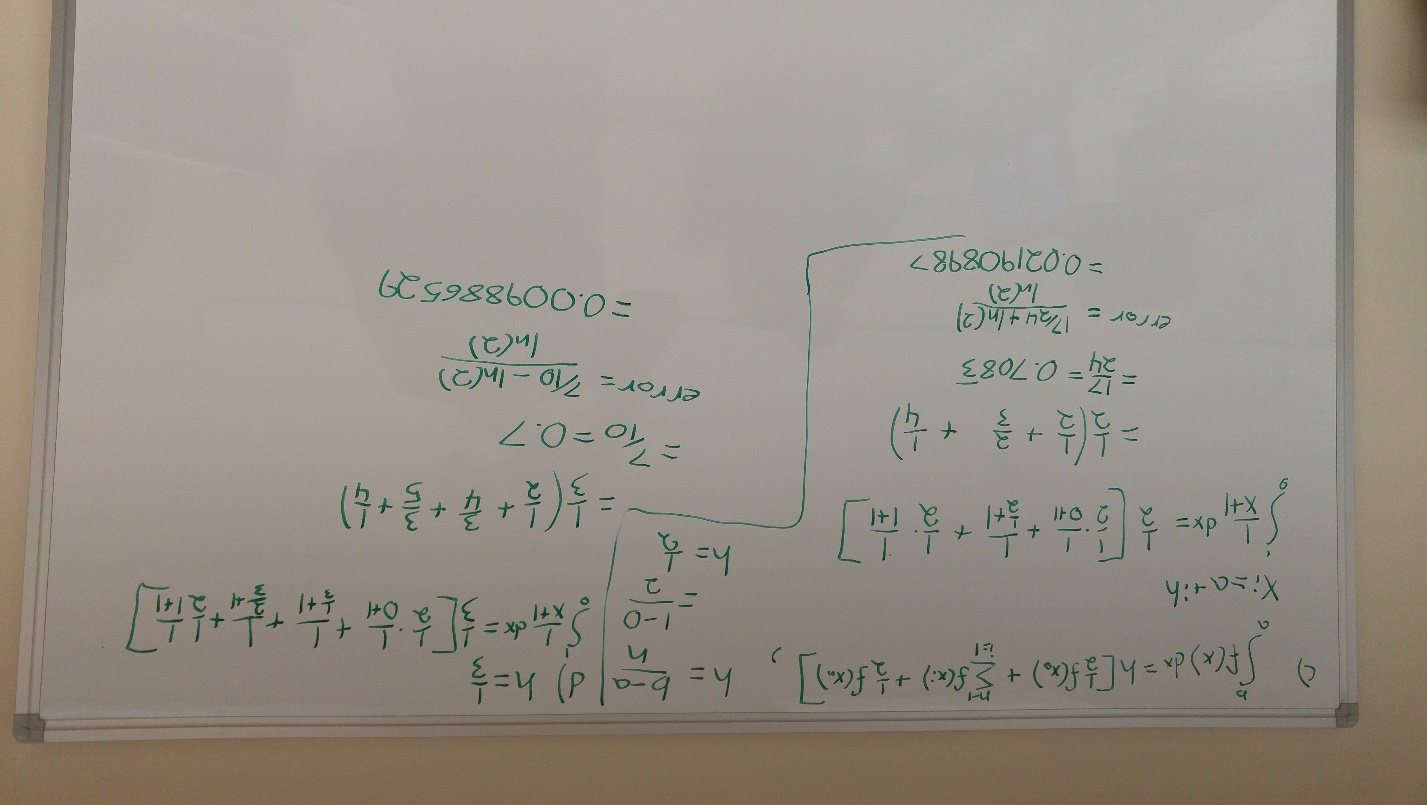


Using the forward derivative formula, we find we have an error of h.

Using the central difference formula, we find we have no error present.

1. 





e) error <= 0.001 n = 1 h = 1 e = 0.082021281 e/h^2 = 0.082

n = 2 h = 1/2 e = 0.021908987 e/h^2 = 0.088

n = 3 h = 1/3 e = 0.009886529 e/h^2 = 0.089

Eh ~ 0.089h^2 🡪 0.089h^2 <= 0.001 or h <= 0.1059998

n = (1 – 0)/h 🡪 n >= 9.434

therefore we can say n = 10 will grant us a error smaller than 0.001.