**Peer Review:**

Reviewer Name: Alejandra Valenzuela

1. What was the function supposed to be called?

falsePosition - correct

1. What is the function supposed to do?

The function was supposed to find the root of a function given some constraints.

1. If inputs and outputs were specified by the assignment, what should the first line of the function file look like?

function [root,fx,ea,iter]=falsePosition(func,xl,xu,es,maxit,varargin)

Before you look at the code:

1. Investigate the help text by typing

>> help function\_name

Does the help text adequately explain the function?

The help text is definitely very clear and concise. It fully explains what the function does, what each variable means, and clarifies any restrictions.

1. Is there something that you would have changed or added?

No, the help text explained everything in an efficient and concise manner.

1. Try and run the algorithm. Consider finding the lowest positive root of the mathematical function: Do you get the right answer?

No, an error message appeared due to exceeding the matrix dimensions.

Review the code

1. If you DIDN’T get the right answer output from the function, can you

see why in the code? What is the problem?

The error message read “index exceeds matrix dimensions” which I am not sure if that is a fault on my part.

1. Is the code well commented? Explain / give an example

The function is definitely well commented on almost every line of code.

Ex: oldroot=x1; %saves old approximation for Approximate Percent Error calculation

1. Are the function variables named appropriately or are they confusing?

The function variables are appropriate because they were the names of the required variables.

1. Would it be easy to modify / expand the code based on the clarity and conciseness of the code? Give suggestions / examples. Try and “break” the program.

Yes, the code gives a clear, step by step explanation of what is being done, so it would be very easy to modify it in a certain spot where one would find necessary. For example, on line 27, if x1<x2, that could easily be changed.

1. Call it with something non-sensical. Call it with something that you KNOW shouldn’t work. Does it give a good error? Give an example of what you tried and an error.

I received the error message of an undefined variable when calling “falsepose” which shows it does give a good error message.

1. Look at the main loop / decision in the algorithm. Can you follow it?

Yes, everything was in order and would make sense to someone who understands what needs to be done in order to use the false position method.

1. Does it make sense? Is it well commented and is it easy to follow along?

Yes, there is comments on almost every line of code that explains why that line of code is necessary and what it does.

1. Is there anywhere you can reduce the memory (fewer variables) or computational cost (better loops, calculations) of the algorithm?

The double function is already built in to the code which I believe already makes it faster, and the code includes everything needed to find the root.

1. Any other general comments or suggestions?

The code was really easy to follow, and I believe it would have worked if I had used correct dimensions. Great job.