The Problem:

In this project we explored the numerical inaccuracies that can arise from different methods of calculating the roots of a quadratic equation given by

$$x^2 + bx + 1 = 0 \tag{1}$$

The methods used to calculate the roots of this quadratic are as follows:

Method 1:

$$-\frac{b}{2} + \frac{|b|}{2}\sqrt{1 - \frac{4}{b^2}} \tag{2}$$

Henceforth, this will be addressed as root 1 with the error calculation for this method addressed as error 1. In the code implementation, the variables "root1" and "error1" correspond to root 1 and error 1, respectively.

Method 2:

$$\left(-\frac{b}{2} - \frac{|b|}{2}\sqrt{1 - \frac{4}{b^2}}\right)^{-1} \tag{3}$$

Henceforth, this will be addressed as root 2 with the error calculation for this method addressed as error 2. In the code implementation, the variables "root2" and "error2" correspond to root 2 and error 2, respectively.

Method 3:

A Taylor series approximation to evaluate

$$-\frac{b}{2}\left(1 - \sqrt{1 - \frac{4}{b^2}}\right) \tag{4}$$

Henceforth, this will be addressed as root 3 with the error calculation for this method addressed as error 3. In the code implementation, the variables "root3" and "error3" correspond to root 3 and error 3, respectively.

Coding Implementation:

Before delving into the actual discoveries made during this project, I would like to discuss my coding implementation. Working in Python, I used numpy arrays to test multiple values of 'b' for each method, tailoring each set of 'b' values to the method (more on this in each method's result). I abstracted the concept of a target error in the coding implementation, allowing me to choose any error value and find the corresponding 'b' value for each method. Additionally, I used the scipy package's Taylor series approximation generator for Method 3. At the end, I used matplotlib to graph the resulting figures. The specific code, both in picture and text format, is attached at the end of this paper.

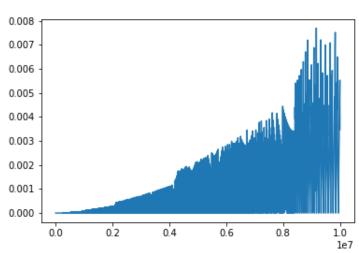
Other Notes:

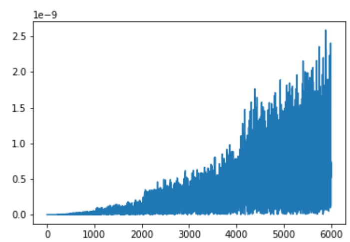
The maximum error permitted for my experiment was chosen to be at 0.001, henceforth referred to as ME (maximum error). This number, though seemingly arbitrary, led to interesting discoveries about the nature of the errors in each of the methods.

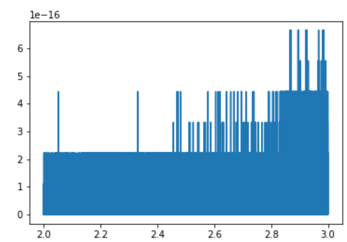
For each method, there is an optimal range of ‘b’ values for which the graph plotting ‘error’ against ‘b’ values actually provides insight. However, since it is also interesting to see how other ranges of ‘b’ impact the error values for other methods, I will include all graphs in the results for each method.

For each method, I worked to find the optimal range of ‘b’ values for which the ME is apparent in the graph. This is why the ranges of ‘b’ values may seem random at first glance and is also the reason why the ranges change for each method.

Results for Method 1:







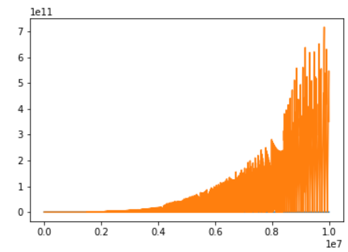
From left to right, the graphs above correspond to Graph 1-a, Graph 1-b, and Graph 1-c.

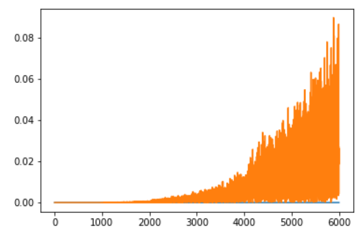
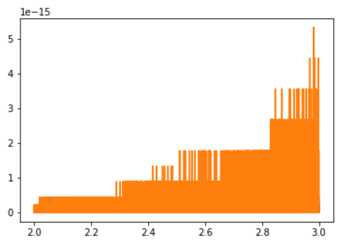
In Graph 1-a, we have chosen the optimal range for ‘b’ values, that being [3, 10000000] with a step of 1, for this method. We are able to see clearly along the y-axis the point at which the ME is crossed. In particular, the b value cutoff for Method 1 for our ME is 6009345, indicating that when ‘b’ is larger than this threshold, the accuracy of the roots approximated by Method 1 is no longer satisfactory.

In Graph 1-b, we have chosen the range of ‘b’ values optimal for Method 2, that being [3,6000] with a step of 1. We can see that the y-axis is on a scale to magnitude 10^{-9}. This means that in this range, Method 1 is too good an approximation to surpass the ME.

In Graph 1-c, we have chosen the range of ‘b’ values optimal for Method 3, that being [2.0001, 3] with a step of .0001. We can see that the y-axis is on a scale to magnitude 10^{-16}. This means that in this range, Method 1 is too good an approximation to surpass the ME. Additionally, the nature of the graph as not smooth and quite blocky indicates that the errors are most likely smaller than 10^{-16}, however Python could not handle smaller numbers.

Results for Method 2:



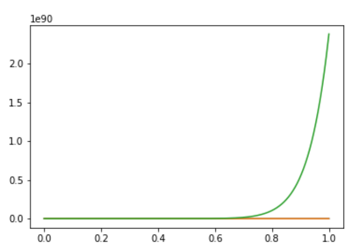
From left to right, the graphs above correspond to Graph 2-a, Graph 2-b, and Graph 2-c.

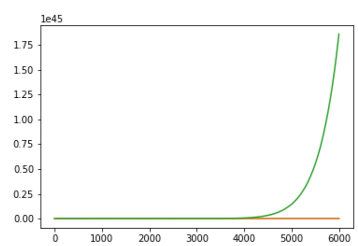
In Graph 2-a, we have chosen the range of ‘b’ values optimal for Method 1, that being [3, 10000000] with a step of 1. We can see that the y-axis is on a scale to magnitude 10^{11}. This means that in this range, Method 2 is too poor an approximation and has already surpassed the ME. However, we cannot clearly tell in the graph where the ME was surpassed, so the range of ‘b’ values is too large.

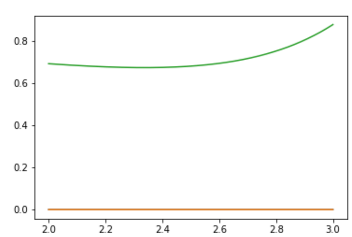
In Graph 2-b, we have chosen the optimal range for ‘b’ values, that being [3, 6000] with a step of 1, for this method. We are able to see clearly along the y-axis the point at which the ME is crossed. In particular, the b value cutoff for Method 1 for our ME is 4238, indicating that when ‘b’ is larger than this threshold, the accuracy of the roots approximated by Method 2 is no longer satisfactory.

In Graph 2-c, we have chosen the range of ‘b’ values optimal for Method 3, that being [2.0001, 3] with a step of .0001. We can see that the y-axis is on a scale to magnitude 10^{-15}. This means that in this range, Method 2 is too good an approximation to surpass the ME.

<u>Results for Method 3:</u>







From left to right, the graphs above correspond to Graph 3-a, Graph 3-b, and Graph 3-c.

In Graph 3-a, we have chosen the range of ‘b’ values optimal for Method 1, that being [3, 10000000] with a step of 1. We can see that the y-axis is on a scale to magnitude 10^{90}. This means that in this range, Method 2 is too poor an approximation and has already surpassed the ME. However, we cannot clearly tell in the graph where the ME was surpassed, so the range of ‘b’ values is too large.

In Graph 3-b, we have chosen the range of ‘b’ values optimal for Method 1, that being [3, 6000] with a step of 1. We can see that the y-axis is on a scale to magnitude 10^{45}. This means that in this range, Method 2 is too poor an approximation and has already surpassed the ME.

However, we cannot clearly tell in the graph where the ME was surpassed, so the range of ‘b’ values is too large.

In Graph 3-c, we have chosen the optimal range for ‘b’ values, that being [2.0001, 3] with a step of .0001, for this method. We are able to see clearly along the y-axis the point at which the ME is crossed. Interestingly, the minimum value for error for this method is 0.67 and occurs at a ‘b’ value of near 2.337. This indicates that all ‘b’ values for this Method are unsatisfactory with respect to our ME. For a more meaningful result, it may be useful to choose a more lenient ME value for a Taylor series approximation.

Overall Results:

These graphs and analysis show us that Method 1 is the best method to calculate roots, Method 2 comes in second, and Method 3 does poorly. What is interesting about this is that the Taylor series approximation starts off at a high error relative to Methods 1 and 2, but does not immediately grow exponentially as is seen in Graph 1-a and Graph 2-b. Instead, in Graph 3-c, we see a slight dip that has a local minimum at the point (2.337, 0.67) and then increases exponentially (this growth is seen in both Graph 3-a and Graph 3-b).

To improve the meaning behind the graphs, I could have graphed them on a logarithmic scale. This would have more strictly shown the exponential nature of the growth (esp. graphs ending in an ‘a’) and provided more insight about the methods.