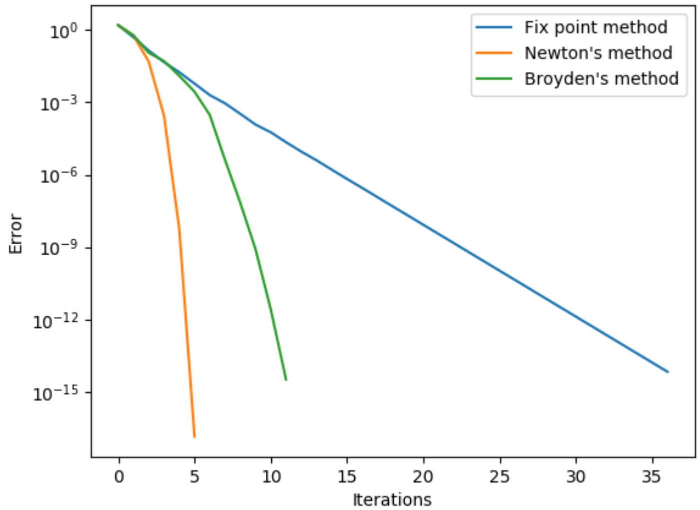
Q2):

The error vs iteration graph shown in the graph below:



The initial guess we choose is [2, 2, 2] since it yields a better result than the other points. The result we get from the fix point method is x1 = 2.99739476e-15, x2 = 3.33333333e-01 and x3 = 3.21433890e-15.From the provided stater code, we have obtian the estimated r and c for the method shown above. For Fix point method, we get r = 0.9990599 and c = 0.4040750. For Newton's method, we have r = 1.8362973 and c = 0.0186061. Last, we have Broyden's methods with r = 1.1666533 and c = 0.1052318. From the above r and c, we can can see that fix point method converges linearly (since r is close to 1 and c < 1). The linear convergence agree with what we see on the plot where the error decent in a straight line. On the other hand, Newton's method converges much faster. From Newton method's r and c apprximations, Newton's method converges quadraticly, which supports what the plot is showing. In the plot, Newton's methods converges in only 5 iterations. Finally, from the approxamtion Broyden's method have a superlinear convergence, which agrees with what we observe on the plot. The convergence speed of Broyden's method is between Fix point method and Broyden's method.

Q3):

The results of the runtime are shown below:

Method	Runtime Mean	Runtime std	Sample num mean	Sample num std
naive	16.827ms	14.640ms	90	79
deflate	20.814ms	21.617ms	25	21
alt deflate	25.046ms	19.310ms	24	17

From the table we observe that the naïve implementation have the fastest runtime of all the other algorithms in the same table. However, if we look at how many sample is drawn, we see that the deflate algorithm and its alternatives draws significantly less sample than the naïve implementation. This means that although the deflate method and its alternative will significatly reduce the number of times to draw, each calculation of f(x) is more computationally expensive than the naïve implementation.