1) Implement gradient descent with BTLS in MATLAB using $\alpha = \{0.1, 0.2, 0.3, 0.4, 0.5\}$ and $\beta = \{0.1, 0.3, 0.5, 0.7, 0.9\}$ with a stopping criterion of $\|\nabla f(x)\| \leq 10^{-4}$. Test your algorithm with the following functions
   a) $f(x_1, x_2) = e^{x_1 + 3x_2} - 0.1 + e^{x_1 - x_3} - 0.1 + e^{-x_1 + 0.1}$
   b) $f(x_1, x_2) = x_1^2 + 10x_2^2$
For each function make a $5 \times 5$ table consisting of the average number of iterations for each pair $(\alpha, \beta)$. To calculate the average number of iterations use 50 random starting points for each pair of $\alpha$ and $\beta$. Plot the graph of $f(x^{(k)}) - p^*$ vs the iteration number $k$ for the starting point $x = [1, -1]^T$ using $\alpha = 0.2$ and $\beta = 0.5$.

2) Implement Newton method in MATLAB. Use the stopping criterion $\lambda^2(x)/2 \leq 10^{-8}$ and the same parameters as in the gradient descent algorithm for BTLS. Test the Newton method as in Problem 1 by answering the same question.

3) Apply CVX to solve the previous problems for proving the correctness of your algorithm (and for finding $p^*$).