MTH 451/EGR 551 - Exam 1
Due Monday March 1, 2021 at 5 p.m.
You are not allowed to collaborate with anyone for this exam.
Please submit all computer codes you developed to solve these problem.

1. **[20 points]** Assume that the scalar function $f(x)$, where $x \in \mathcal{R}^n$, has continuous derivatives.

   (a) Find a *forward difference* approximation for $df(x)/dx$ that has a truncation error that is of order h^2. (Hint: Use Taylor's theorem to write $f(x+h)$ and $f(x+2h)$ as polynomials centered at x for some increment h.)

   (b) Find an estimate of the increment h that will minimize the roundoff and truncation errors.

   (c) Use the formula you have derived to write a MATLAB/Octave procedure to compute the gradient of a function. Use your MATLAB/Octave code to compute the derivatives of the function
   $$f(x_1, x_2) = (4x_1^2 + 2x_2^2 + 4x_1x_2 + 2x_2 + 1)e^{x_1}$$
   at $x_1 = -1, x_2 = 1$, and $x_1 = 10, x_2 = 10^{-3}$. What are the relative and absolute errors in each case?

2. **[20 points]** Matrix inverse.

   (a) (Forward Substitution) Given a nonsingular lower triangular matrix $L \in \mathcal{R}^{n\times n}$ and a matrix $B \in \mathcal{R}^{n\times m}$. Develop an algorithm to solve the linear equation $LY = B$, where $Y \in \mathcal{R}^{n\times m}$. Implement the algorithm in MATLAB/Octave such that the input to the function is L and B and the output is Y.

   (b) (Backward Substitution) Given a nonsingular upper triangular matrix $U \in \mathcal{R}^{n\times n}$ and a matrix $Y \in \mathcal{R}^{n\times m}$. Develop an algorithm to solve the linear equation $UX = Y$, where $X \in \mathcal{R}^{n\times m}$. Implement the algorithm in MATLAB/Octave such that the input to the function is U and Y and the output is X.

   (c) Using the forward substitution algorithm and backward substitution algorithm developed above. Develop and implement an LU factorization algorithm to compute the inverse of a nonsingular matrix $A \in \mathcal{R}^{n\times n}$ by solving the linear system $AX = I$, where $I \in \mathcal{R}^{n\times n}$ is the identity matrix. The input to the function is A and the output is X. You can use the LU factorization algorithm developed in class, i.e., `LUsolve.m`.

   (d) Use your algorithm to find the inverse of the matrix H with elements
   $$h_{ij} = 1/(i+j-1),\ i = 1, 2, \cdots, n, \quad j = 1, 2, \cdots, n.$$
   for cases where $n = 5$ and $n = 13$. In each case compute and print the values $\|HX - I\|_\infty$, where X is the computed inverse of H and I is the identity matrix.

3. **[25 points]** The steepest descent technique, *with exact line searches*, for minimizing the scalar function $f(x)$ where $x \in \mathcal{R}^n$ is as follows. Given: x^0 (an initial estimate of the solution), ϵ_1 (a convergence tolerance),

   Step 1: Compute $d = -\nabla f(x^0)$, if $\|d\| \leq \epsilon_1$, stop, x^0 is a stationary point.

Step 2: Find an expression for the α that minimizes $f(x + \alpha d)$. (That is find α so that $df(x + \alpha d)/d\alpha = 0$.)

Step 3: Set $x^0 = x^0 + \alpha^* d$, where α^* is the minimum computed in Step 2. Go to Step 1.

Specialize this algorithm to minimize the function

$$f(x) = \frac{1}{2} x^T Q x - b^T x \tag{1}$$

where Q is a symmetric positive definite matrix. This should be done as follows;

(a) Using (1) derive an analytical expression for $d = -\nabla f(x)$.

(b) Find the α that minimizes $f(x + \alpha d)$, i.e., perform an exact line search. (**Do not use the back-tracking line search**.)

(c) Write a MATLAB/Octave program to implement the algorithm with d and α determined in parts (a) and (b).

(d) Use your program to find the minimum of the function if

$$Q = \begin{bmatrix} 1 & 5 & 6 & 7 & 8 \\ 5 & 26 & 31 & 36 & 41 \\ 6 & 31 & 38 & 44 & 50 \\ 7 & 36 & 44 & 52 & 59 \\ 8 & 41 & 50 & 59 & 68 \end{bmatrix}, \quad b = \begin{bmatrix} 65 \\ 335 \\ 406 \\ 475 \\ 540 \end{bmatrix}, \quad \epsilon_1 = 1.0 \times 10^{-4}, \quad x^0 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Plot $\|d\|_\infty$ versus the iteration number.

Hint: This problem requires a large number of iterations (approximately 5000) to converge.

4. [**35 points**] Write a MATLAB/Octave function that implements the function minimization algorithm described below. The input to the MATLAB/Octave procedure is the function to minimize, the initial solution estimate and a convergence tolerance.

Input: An initial estimate x, and a convergence tolerance $\epsilon_t > 0$.
Output: x such that $\|\nabla f(x)\| \le \epsilon_t$.

$d = -\nabla f(x)$
for $k = 0, 1, \cdots$ **do**
  if $\|\nabla f(x)\| \le \epsilon_t$, break
  Find $\alpha = \alpha^* > 0$ that approximately minimizes $f(x + \alpha d)$ via the Armijo rule
  $\bar{x} = x + \alpha^* d$
  $d = -\nabla f(\bar{x})$
  $s = \bar{x} - x$
  $y = \nabla f(\bar{x}) - \nabla f(x)$
  if $s^T y > 0$
    $d = \left(d + \left(1 + \frac{y^T y}{s^T y}\right) \frac{s s^T d}{s^T y} - \left(\frac{s y^T d + y s^T d}{s^T y} \right) \right)$
  end if
  $x = \bar{x}$
end for

- Use the implementation of the function minimization algorithm to find the minimum of the function

$$V(\theta) = \sum_{i=1}^{N-1} m_i g h_i + \frac{\rho}{2}\left(\left(\mu L - \sum_{i=1}^{N} l_i \cos\theta_i\right)^2 + \left(\sum_{i=1}^{N} l_i \sin\theta_i\right)^2\right),$$

  where $\theta = [\theta_1, \theta_2, \ldots, \theta_N]^T$, $h_i = \sum_{j=1}^{i} l_j \sin\theta_j$. (Notice that $h_i = h_{i-1} + l_i \sin\theta_i$ with $h_0 = 0$.) Here $N = 11$, $\rho = 10^3$, $g = 9.8$, $\mu = 0.7$, $L = 1$, $m_i = 1/(N-1)$, $i = 1, 2, \ldots, N-1$. $l_i = L/N$, $i = 1, 2, \ldots, N$ and convergence tolerance $\epsilon_t = 10^{-4}$.
- Find θ that minimizes $V(\theta)$ using the minimization algorithm implemented.
- Print the value of θ_1 and θ_N. Plot θ.
- Compute $x_i = \sum_{j=1}^{i} l_j \cos\theta_j$, $y_i = \sum_{j=1}^{i} l_j \sin\theta_j$, $i = 1, 2, \ldots, N$. Print the following values x_1, y_1, x_N and y_N. Plot x versus y.
- Attempt to find solutions when $N = 101$ and $N = 1001$. For each value of N print the value of $\theta_1, \theta_N, x_1, y_1, x_N$ and y_N. Plot θ and plot x versus y.

Hint: Try to solve this problem with small N and $\epsilon_t = 10^{-3}$ first.