HOMEWORK 4

ASSIGNED: 10/02/25

DUE: 10/09/25 (on ELC, PDF and MATLAB), before class

Revisiting your shape functions for 1D, create a lookup function for both **N** and **GN** in two dimensions.

N = N2D(xi,eta,nn)

GN = GN2D(xi,eta,nn)

Where *xi* and *eta* are the location in the parent domain within the element both ranging from -1 to 1 and *nn* is the number of nodes within the element. *nn* should support 4 and 9 nodes.

To test these functions you will create plots of the following equations. Create evenly spaced points in the specified domain for your nodal coordinates (you may use `xiplot = linspace(-1,1,10);` `etaplot = linspace(-1,1,10);` if you like). Using your shape functions, create plots of the function using both the linear (4 nodes) and quadratic (9 nodes) solutions. Plot both the function and the gradient of the function as defined by:

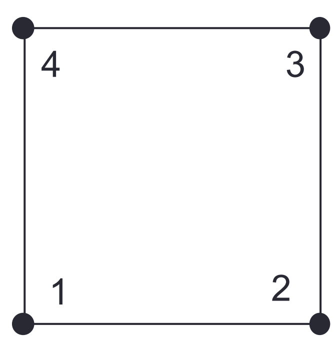
$$\theta^e(\xi,\eta) = [N]^e [d]^e$$

$$\nabla\theta^e(\xi,\eta) = [GN]^e [d]^e$$

Case 1 (linear shape functions): The coordinates for each node (4 total) are provided as rows within $[\xi,\eta]^e$. Corresponding nodal values for θ are provided as each row in $[d]^e$.

$$\theta(\xi,\eta) = \xi + \eta$$

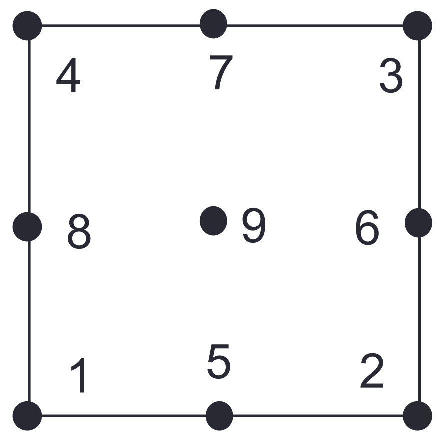
$$[\xi,\eta]^e = \begin{bmatrix} -1 & -1 \\ 1 & -1 \\ 1 & 1 \\ -1 & 1 \end{bmatrix} \quad [d]^e = \begin{bmatrix} -2 \\ 0 \\ 2 \\ 0 \end{bmatrix}$$



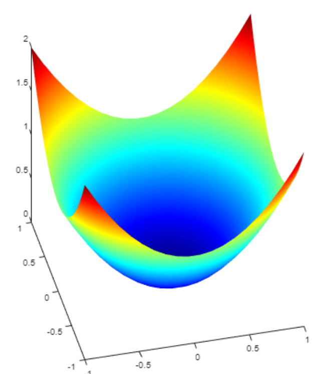
Case 2 (quadratic shape functions): The coordinates for each node (9 total) are provided as rows within $[\xi,\eta]^e$. Corresponding nodal values for θ are provided as each row in $[d]^e$.

$$\theta(\xi,\eta) = \xi^2 + \eta^2$$

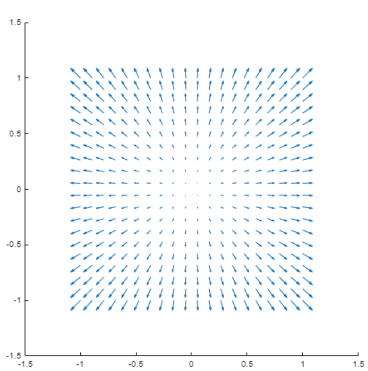
$$[\xi,\eta]^e = \begin{bmatrix} -1 & -1 \\ 1 & -1 \\ 1 & 1 \\ -1 & 1 \\ 0 & -1 \\ 1 & 0 \\ 0 & 1 \\ -1 & 0 \\ 0 & 0 \end{bmatrix} \quad [d]^e = \begin{bmatrix} 2 \\ 2 \\ 2 \\ 2 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$



For plotting the behaviors, I recommend the following:



Plotting the function: Use the **surf** command. Set up a nested for loop, where every single value is stored in a matrix using dual indices i,j. Each point will need a coordinate in ξ, a coordinate in η, and a value for θ at this point in ξ and η. If you use 10 points in ξ and η then this will create 10x10 matrices of values for each coordinate and θ. As an example, you should produce something like the figure to the right.



Plotting the derivatives. Since this is the gradient, this will be a vector plot. Use the **quiver** command in MATLAB. This is similar to the **surf** command, except now you will store the values for $\delta\theta/\delta\xi$ and $\delta\theta/\delta\eta$ to create the two components of the vector at each point in ξ and η. Rather than accepting a single value to plot at each coordinate, you now provide two which define the direction of the vector. You will produce something similar to the figure to the left.