**Richard Clottey**

**Description of Solution**

In computing linearization of the lower triangular matrix, the rows and columns were specified with the variables I and j respectively.

A nested loop consisting of two for loops will loop through the square matrix and if the current row number is less than the column number, 0 is inserted in that position.

For the row numbers that were greater than the column numbers, the random numbers were generated in those positions for each iteration of the for loop.

F(), which computes I = F(I, j), calculates the integer position of a one dimensional array by using the indices of the multidimensional array of the lower triangular matrix.

This is computed as **I = i \* col + j,** where I represents the index of the one-dimensional array, i represents the row index of the lower triangular matrix, j represents the column index of the lower triangular matrix as well and col represents the total number of columns of the lower triangular matrix.

The inverse F-1(I) is computed as **i = I / col** and **j = I % col** where I represents the index of the one-dimensional array, i represents the new row for the new matrix and j represents the new column for the new matrix. Col also represents the total number of columns of the new matrix as well.