Problem Set 2

Due Date: March 25

Write a code that uses Givens rotations to bring a given matrix A to the upper Hessenberg form. Recall that if $(x, y)^T$ is a two-dimensional real vector, the corresponding (two-dimensional) Givens rotation is a 2×2 rotation Q such that the second coordinate of $Q \cdot (x, y)^T$ is zero. An $n-$dimensional Givens rotation fixes $n-2$ coordinates and applies a 2-dimensional Givens rotation on the other 2 coordinates. Recall that an upper Hessenberg matrix is a matrix having zero entries below the first subdiagonal.

In other words, given an $n \times n$ matrix A, you need to produce an $n \times n$ orthogonal transformation U such that

$$U \cdot A \cdot U^T = B, \tag{1}$$

where B is an upper Hessenberg matrix, and U is a composition of Givens rotations.

In FORTRAN, your calling sequence should be

$$upperhes(a, n, u, b) \tag{2}$$

where

$a(n, n)$ is a (real) $n \times n$-matrix to be transformed (input parameter)

n is the (integer) size of the matrix (input parameter)

$u(n, n)$ is a (real) $n \times n$-matrix (output parameter). The matrix u is orthogonal.

$b(n, n)$ is a (real) $n \times n$-matrix (output parameter). The matrix b is an upper Hessenberg matrix satisfying $b = uau^T$.

In C, your calling sequence should be

$$void\ upperhes(double*\ a, int\ n, double*\ u, double*\ b), \tag{3}$$

where

a points to an array of doubles of size n^2, containing $a(1,1), a(1,2), \ldots, a(1,n), a(2,1), \ldots, a(n,n)$, a being the $n \times n$ matrix to be transformed (input parameter)

n is the (integer) size of the matrix (input parameter)

u points to an array of doubles of size n^2, containing $u(1,1), u(1,2), \ldots, u(1,n), u(2,1), \ldots, u(n,n)$, u being the $n \times n$ orthogonal matrix. u is a composition of Givens rotations utilized to bring a into the upper Hessenberg form (output parameter, memory allocated by the user)

b points to an array of doubles of size n^2, containing $b(1,1), b(1,2), \ldots, b(1,n), b(2,1), \ldots, b(n,n)$, b being the $n \times n$ matrix. The matrix b is an upper Hessenberg matrix satisfying $b = uau^T$ (output parameter, memory allocated by the user)

REMARK 1. Make sure that the calling sequence of your function is exactly as specified in the assignment.

REMARK 2. Test your code before submission. In particular, make sure that the matrix u is indeed orthogonal and b is indeed an upper Hessenberg matrix satisfying $b = uau^T$.

REMARK 3. If the matrix a is symmetric, then b should be tridiagonal. You can use this fact to test your code.