Problem 1.

Solution: By the spectral theorem, since A is a normal matrix, this implies that

$$A = U\Lambda U^*$$

where U is an unitary matrix. Thus, we can see that

$$A^* = U\Lambda^* U^*$$

where we can see that A has the same eigenvectors but conjugate eigenvalues as A^*. $\square$

Problem 2.

Solution:

```
% Compute runtime and spectral norm of symmetric random matrices

% List of matrix sizes to test
sizes = [36 , 50, 100, 200, 400, 500, 600, 700, 800, 1000, 1100,
   1200, 1400, 1450, 1500];

numSizes = numel(sizes);
runtimes = zeros(1, numSizes);
spectralNorms = zeros(1, numSizes);

for k = 1:numSizes
    n = sizes(k);

    % Generate random symmetric matrix
    A = randn(n);
    A = (A + A')/2;

    % Time the eigenvalue computation
    tic;
    ev = eig(A);    % for a symmetric A, eig uses a symmetric solver
    runtimes(k) = toc;

    % Record the spectral norm (max absolute eigenvalue)
    spectralNorms(k) = max(abs(ev));
end

% Plot runtime vs. matrix size
figure;
plot(sizes, runtimes, '-o', 'LineWidth', 1.5, 'MarkerSize', 8);
xlabel('Matrix size n');
ylabel('Runtime (seconds)');
title('Runtime of Eigenvalue Decomposition vs. Matrix Size');
grid on;

% Plot spectral norm vs. matrix size
figure;
plot(sizes, spectralNorms, '-o', 'LineWidth', 1.5, 'MarkerSize', 8);
xlabel('Matrix size n');
ylabel('Spectral Norm (max |\lambda|)');
title('Spectral Norm vs. Matrix Size');
grid on;
```

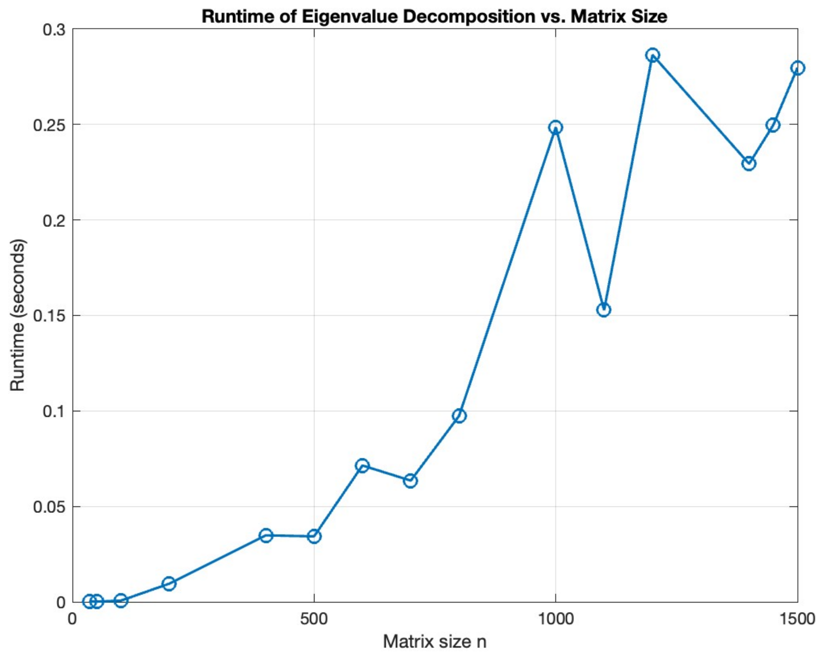

Figure 1: Runtime of Eigenvalue versus Matrix Size

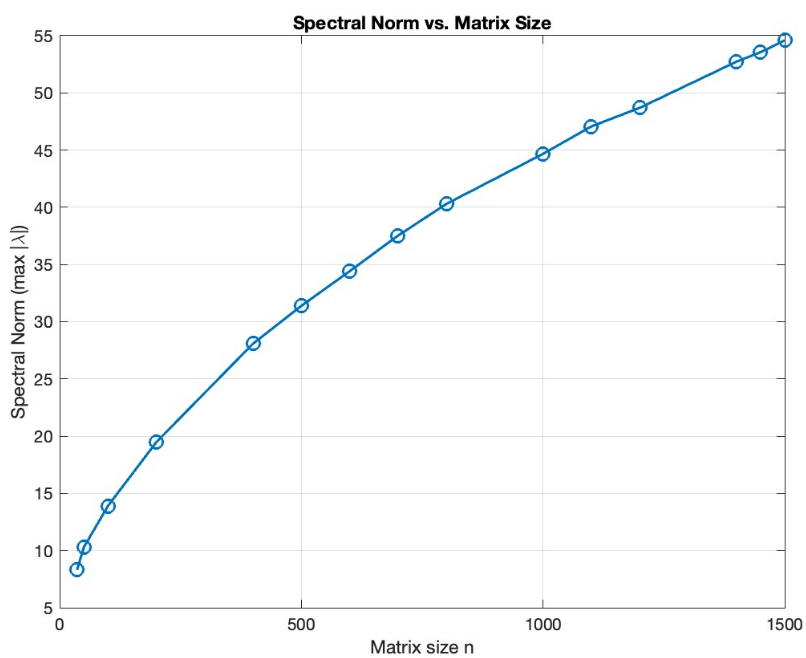

Figure 2: Spectral Norm versus Matrix Size

The run time of the Eigenvalue decompoistion seems to be $\mathcal{O}(n^3)$ and the spectral norm seems to be $\mathcal{O}(\sqrt{n})$ $\square$

Problem 3.

Solution: Since U and V are orthonormal, they are basis for the subspace in $\mathbb{R}^3$, which in this case is a 2-d plane. By definition, the angle between two subspaces are defined by the smallest angle between any two vectors, where one comes from each respective subspace. Thus, we can proceed to with the calculation of these principal angles. To calculate these angles, we must calculate the inner product of between each vector in the basis, or $U^T V$, where we take the SVD to reveal that:

$$U^T V = J \Sigma P^T$$

where J denotes the left singular matrix and P represents the right singular matrix. Each respective singular value of $U^T V$ represents the cosine of each respective value. Given the orthonormal nature of U^T and V, we can see that just taking the diagnol values will suffice in calcualting the principal angles. Thus, we can see that:

$$\theta_i = \cos^{-1}((U^T V)_{ii})$$

$\square$

Problem 4.

Solution:

```
  A = [17 22 27 32;
     22 29 36 43;
     27 36 45 54;
     32 43 54 65];

dim_A = size(A);

assert(dim_A(1) == dim_A(2))

A_k_vec = cell(dim_A(1),1);

for k = 1:dim_A(1)
    A_R = A(1:k, :);
    A_C = A(:, 1:k);
    A_U = A(1:k, 1:k);
    det_A_U = det(A_U);

    if det_A_U == 0
        disp("Matrix is not invertible at k = " + k)
    else
        A_k_vec{k} = A_C * inv(A_U) * A_R;
    end
end

for k = 1:length(A_k_vec)
    if ~isempty(A_k_vec{k})
        disp("k = " + k)
        disp(A_k_vec{k})
    end
end
```

Thus, we can see that when $k = 3, 4$ we are dealing with nonsingular matrices. Hence,

$$A_1 = \begin{bmatrix} 17.0000 & 22.0000 & 27.0000 & 32.0000 \\ 22.0000 & 28.4706 & 34.9412 & 41.4118 \\ 27.0000 & 34.9412 & 42.8824 & 50.8235 \\ 32.0000 & 41.4118 & 50.8235 & 60.2353 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 17.0000 & 22.0000 & 27.0000 & 32.0000 \\ 22.0000 & 29.0000 & 36.0000 & 43.0000 \\ 27.0000 & 36.0000 & 45.0000 & 54.0000 \\ 32.0000 & 43.0000 & 54.0000 & 65.0000 \end{bmatrix}$$

Note the complexity of forming $\mathcal{O}(n^3)$, as we consisttantly use Guassian Elimination, (we proved in class that it is $\mathcal{O}(n^3)$) $\square$