Question 2

Code

```
function A2Q2()
clc

NN = [64,128,256,512,1024,2048,4096];
% taking k as half the square root of the matrix: Case 1
kk = NN/2;

% taking k as the square root of the matrix Case 2
%kk = NN;

for II = 1:length(NN)

    % Defining descrete Laplacian
    N = NN(II);
    A = Laplacian(N);

    % Defining eigenvalues of descrete Laplacian
    Eigen = @(ii,jj,N) 4-2*(cos(ii*pi/(N+1))+cos(jj*pi/(N+1)));
    SmallestEigA = [Eigen(1,1,N);Eigen(2,1,N);Eigen(1,2,N)];
    LargestEigA = [Eigen(N,N,N);Eigen(N-1,N,N);Eigen(N,N-1,N)];

    b = randn(N^2,1);

    % Calling Lanczos code
    k = kk(II);
    [T] = lancz(A, b, k);

    % Calculating eigenvalues of T
    OPTS.maxit = 1e6;
    SmallestEigT = eigs(T,3,'SM',OPTS);
    LargestEigT = eigs(T,3,'LM',OPTS);

    % Sort eigenvalues
    SSEA = sort(SmallestEigA);
    SLEA = sort(LargestEigA);
    SSET = sort(SmallestEigT);
    SLET = sort(LargestEigT);

    % Defining table data
    data = [SSEA,SSET,abs(SSEA-SSET),SLEA,SLET,abs(SLEA-SLET)];

    % Set up some options
    tblOpts = {'header',{'Smallest Eig A','Smallest Eig T',...
        'inf-norm error','Largest Eig A','Largest Eig T'...
        ,'inf-norm error'},'format',{'%1.4e','%1.4e','%1.4e'...
        ,'%1.6f','%1.6f','%1.4e'},'align','center','delim','|',...
        'printRow',true};
```

```
    for ii = 1:size(data,1);
        table(['Table of Eigenvalues for n = ',num2str(NN(II)^2),...
            ' and k = ',num2str(kk(II))],data(1:ii,:),tblOpts{:}...
            ,'finalRow',ii == size(data,1));
    end

end


    function [A] = Laplacian(n)
        % Creating discretised Laplacian

        e = ones(n,1);

        % Creating sparse diagonal matrices
        I = spdiags(e,0,n,n);
        I1 =spdiags(e,1,n,n);
        I2 = spdiags(e,-1,n,n);

        % Creating 1D Convection-Diffusion matricies
        A1D = 2*I -1*I1 - 1*I2;

        % Creating 2D Convection-Diffusion matrix
        A = kron(I,A1D)+kron(A1D,I);

    end



    function [T,Q] = lancz(A, b, k)
        %function [T,Q] = lancz(A, b, k)
        %
        % Function the performs the Lanczos process
        %
        % Input:
        %        A - Symmetic matrix
        %        b - initial guess
        %        A - number of steps in the Lanczos algorithm
        %
        % Output:
        %        T - Symmetic Hessenberg matrix (Tridiagonal)
        %        Q - (OPTIONAL) orthogonal basis

        n = length(b);
        qprev = sparse(n,1);
        q = b / norm(b);
        beta = [];
        alpha = [];

        if nargout == 2
            Q = [];
```

```
        end

        for i = 1:k
            v = A*q;
            alpha(i) = q' * v;

            if i == 1
                v = v - alpha(i)*q;
            else
                v = v - beta(i-1)*qprev - alpha(i)*q;
            end
            beta(i) = norm(v);
            qprev = q;

            if nargout == 2
                Q = [Q,q];
            end

            if (abs(beta(i)) < 1e-10)
                break
            end
            q = v / beta(i);
        end
        beta = beta(:);
        T = spdiags([beta alpha(:) [0;beta(1:end-1)]],[-1:1],i,i);
    end


end
```

Case 1: $\mathbf{k} = \frac{\sqrt{n}}{2}$

Table of Eigenvalues for n = 4096 and k = 32

Smallest Eig A	Smallest Eig T	inf-norm error	Largest Eig A	Largest Eig T	inf-norm error
4.6711e-03	9.9835e-03	5.3124e-03	7.988328	7.858289	1.3004e-01
1.1672e-02	6.3999e-02	5.2327e-02	7.988328	7.938795	4.9533e-02
1.1672e-02	1.4837e-01	1.3670e-01	7.995329	7.984305	1.1024e-02

Table of Eigenvalues for n = 16384 and k = 64

Smallest Eig A	Smallest Eig T	inf-norm error	Largest Eig A	Largest Eig T	inf-norm error
1.1861e-03	2.6827e-03	1.4966e-03	7.997035	7.960694	3.6341e-02
2.9649e-03	1.6376e-02	1.3411e-02	7.997035	7.985630	1.1405e-02
2.9649e-03	3.8718e-02	3.5753e-02	7.998814	7.997029	1.7853e-03

Table of Eigenvalues for n = 65536 and k = 128

Smallest Eig A	Smallest Eig T	inf-norm error	Largest Eig A	Largest Eig T	inf-norm error
2.9885e-04	1.7556e-03	1.4568e-03	7.999253	7.989690	9.5626e-03
7.4711e-04	4.7625e-03	4.0154e-03	7.999253	7.995436	3.8167e-03
7.4711e-04	9.6660e-03	8.9188e-03	7.999701	7.998868	8.3270e-04

Table of Eigenvalues for n = 262144 and k = 256

Smallest Eig A	Smallest Eig T	inf-norm error	Largest Eig A	Largest Eig T	inf-norm error
7.5006e-05	2.4483e-04	1.6982e-04	7.999812	7.997691	2.1218e-03
1.8751e-04	1.0464e-03	8.5886e-04	7.999812	7.999091	7.2112e-04
1.8751e-04	2.1647e-03	1.9772e-03	7.999925	7.999749	1.7558e-04

Table of Eigenvalues for n = 1048576 and k = 512

Smallest Eig A	Smallest Eig T	inf-norm error	Largest Eig A	Largest Eig T	inf-norm error
1.8788e-05	3.6242e-05	1.7454e-05	7.999953	7.999438	5.1471e-04
4.6970e-05	2.5186e-04	2.0489e-04	7.999953	7.999727	2.2568e-04
4.6970e-05	6.1874e-04	5.7177e-04	7.999981	7.999936	4.5595e-05

Table of Eigenvalues for n = 4194304 and k = 1024

Smallest Eig A	Smallest Eig T	inf-norm error	Largest Eig A	Largest Eig T	inf-norm error
4.7016e-06	9.9201e-06	5.2185e-06	7.999988	7.999844	1.4389e-04
1.1754e-05	6.2858e-05	5.1104e-05	7.999988	7.999929	5.9442e-05
1.1754e-05	1.4510e-04	1.3334e-04	7.999995	7.999984	1.0981e-05

Table of Eigenvalues for n = 16777216 and k = 2048

Smallest Eig A	Smallest Eig T	inf-norm error	Largest Eig A	Largest Eig T	inf-norm error
1.1760e-06	4.8539e-06	3.6779e-06	7.999997	7.999961	3.5709e-05
2.9399e-06	1.8653e-05	1.5713e-05	7.999997	7.999983	1.3771e-05
2.9399e-06	3.6771e-05	3.3831e-05	7.999999	7.999996	2.3477e-06

Case 1: k = $\sqrt{n}$