ESERCIZI DI CALCOLO NUMERICO

Esercizio 1. Si consideri la matrice tridiagonale $A_n \in \mathbb{R}^{n\times n}$ definita da

$$A = \begin{bmatrix} 0 & 2 & & \\ 1 & 0 & \ddots & \\ & \ddots & \ddots & 2 \\ & & 1 & 0 \end{bmatrix}.$$

1. Sia $D = \mathrm{diag}(1, 2, 4, \ldots, 2^{n-1})$. Si determini $B_n = D^{1/2} \cdot A_n \cdot D^{-1/2}$.
2. Si dimostri che A_n ha autovalori reali.
3. Per $n \geq 2$ si dica se A_n ammette fattorizzazione LU.
4. Si dimostri che $A_n + 4I_n$ è invertibile per ogni $n \geq 1$.
5. Determinare un insieme di inclusione per gli autovalori di $A_n + 4I_n$, $n \geq 1$.
6. Analizzare sperimentalmente il costo computazionale della risoluzione di un sistema lineare $(A_n + 4I_n)\boldsymbol{x}_n = \boldsymbol{b}_n$ dove $\boldsymbol{b}_n$ è un vettore generato casualmente mediante la funzione `rand` ed il sistema lineare è risolto mediante la routine `tridsolve.m` riportata di seguito.

```
function x = tridisolve(a,b,c,d)
%   TRIDISOLVE  Solve tridiagonal system of equations.
%     x = TRIDISOLVE(a,b,c,d) solves the system of linear equations
%     b(1)*x(1) + c(1)*x(2) = d(1),
%     a(j-1)*x(j-1) + b(j)*x(j) + c(j)*x(j+1) = d(j), j = 2:n-1,
%     a(n-1)*x(n-1) + b(n)*x(n) = d(n).
%
%   The algorithm does not use pivoting, so the results might
%   be inaccurate if abs(b) is much smaller than abs(a)+abs(c).
%   More robust, but slower, alternatives with pivoting are:
%     x = T\d where T = diag(a,-1) + diag(b,0) + diag(c,1)
%     x = S\d where S = spdiags([[a; 0] b [0; c]],[-1 0 1],n,n)

%   Copyright 2014 Cleve Moler
%   Copyright 2014 The MathWorks, Inc.

x = d;
n = length(x);

for j = 1:n-1
   mu = a(j)/b(j);
   b(j+1) = b(j+1) - mu*c(j);
   x(j+1) = x(j+1) - mu*x(j);
end
```

```
x(n) = x(n)/b(n);
for j = n-1:-1:1
   x(j) = (x(j)-c(j)*x(j+1))/b(j);
end
```