

Specifiche per la costruzione dei codici per la soluzione dei sistemi lineari con i metodi iterativi basati sullo splitting della matrice

1. Costruire una **m-function jacobi.m**, che, presa

in input

- la matrice **A**,
- il termine noto **b**,
- la soluzione iniziale **x0**,
- il numero di iterazioni massime **maxiters**
- la precisione richiesta, **prec**

risolva il sistema lineare $Ax=b$, con il metodo iterativo di Jacobi e restituisca

in output

- la soluzione calcolata **x**,
- il vettore **err** con gli errori relativi ad ogni iterazione,
- il numero di iterazioni **k** necessarie per calcolare la soluzione con la precisione richiesta,
- il raggio spettrale **rho** della matrice di iterazione.

```
[x,err,k,rho]=jacobi(A,b,x0,maxiters,prec);
```

2. Costruire una **m-function gauss_seidel.m**, che, presa

in input

- la matrice **A**,
- il termine noto **b**,
- la soluzione iniziale **x0**,
- il numero di iterazioni massime **maxiters**
- la precisione richiesta, **prec**

risolva il sistema lineare $Ax=b$, con il metodo iterativo di Gauss-Seidel e restituisca

in output

- la soluzione calcolata x ,
- il vettore **err** con gli errori relativi ad ogni iterazione,
- il numero di iterazioni **k** necessarie per calcolare la soluzione con la precisione richiesta,
- il raggio spettrale **rho** della matrice di iterazione.

```
[x,err,k,rho]=gauss_seidel(A,b,x0,maxiters,prec);
```

Suggerimenti per l'implementazione di ciascuna delle due m-function richieste.

Costruire a partire dalla matrice A

- la matrice diagonale **D** avente come elementi gli elementi diagonali di A (usare l'istruzione matlab **D=diag(diag(A));**
- La matrice E ottenuta da A considerando solo il triangolo inferiore esclusi gli elementi diagonali, usare l'istruzione matlab **E=tril(A,-1);**
- La matrice F ottenuta da A considerando solo il triangolo superiore esclusi gli elementi diagonali, usare l'istruzione matlab **F=triu(A,1);**

Costruire le matrici della decomposizione M ed N, caratteristici di ciascuno dei tre metodi considerati, Jacobi, Gauss-Seidel, Gauss-seidel con rilassamento;

Jacobi $\rightarrow M=D, N=-(E+F)$

Gauss_Seidel $\rightarrow M=D+E, N=-F$

Gauss-Seidel con rilassamento $\rightarrow M=D+\omega E; N=((1-\omega)*D-\omega F);$

Costruire la matrice di Iterazione $T = \text{inv}(M) * N$;

Calcolare il raggio spettrale di T, cioè l'autovalore di modulo massimo, con l'istruzione matlab `rho=abs(eigs(T,1,'lm'))`;

Se $\text{rho} \geq 1$ visualizzare un messaggio di errore ed uscire dal programma.

Instaurare il procedimento iterativo nel seguente modo:

inizializzare

```
arresto=1;  
k=1;  
xold=x0;  
q=inv(M)*b;
```

```
while arresto>=prec && k <= maxiters (questa è la condizione di ingresso nel ciclo)
```

```
    xnew=T*xold+q; (formula di aggiornamento della soluzione)
```

```
    arresto=norm(xnew-xold,1)/norm(xnew,1); (Errore relativo della soluzione)
```

```
    err(k)=arresto;
```

```
    xold=xnew;
```

```
    k=k+1; incremento contatore delle iterazioni
```

```
end
```

k=k-1; numero di iterazioni che sono state effettuate per calcolare la soluzione con la precisione richiesta.