

RELAZIONE LABORATORIO DI CALCOLO

MATTEO SCHIRINZI 20035542

TRACCIA

Elaborare un programma per risolvere un sistema di equazioni lineari con il metodo iterativo di Gauss-Seidel leggendo i coefficienti e i termini noti da file.

I SISTEMI LINEARI

DEFINIZIONE

I sistemi di equazioni lineari sono insiemi di equazioni lineari, dove ogni equazione è una funzione lineare delle stesse variabili. Formalmente, un sistema di equazioni lineari può essere espresso come:

$$S = \{ \begin{aligned} &(a_{11} * x_1 + a_{12} * x_2 + \dots + a_{1n} * x_n = b_1), \\ &(a_{21} * x_1 + a_{22} * x_2 + \dots + a_{2n} * x_n = b_2), \\ &\dots, \\ &(a_{m1} * x_1 + a_{m2} * x_2 + \dots + a_{mn} * x_n = b_m) \end{aligned} \}$$

dove S è il sistema di equazioni lineari, x_1, x_2, \dots, x_n sono le variabili del sistema, a_{ij} rappresenta il coefficiente della variabile x_j nell' i -esima equazione, e b_i rappresenta il termine noto dell' i -esima equazione.

Un sistema di equazioni lineari può essere risolto trovando un insieme di valori delle variabili che soddisfino tutte le equazioni simultaneamente. In generale, il numero di soluzioni possibili dipende dal numero di equazioni e variabili e dalla forma dei coefficienti delle equazioni.

RISOLUZIONE

METODI DIRETTI

I metodi diretti per la risoluzione dei sistemi lineari sono basati sulla manipolazione algebrica del sistema di equazioni per ottenere una soluzione esatta in un numero finito di passi. Questi metodi sono solitamente utilizzati per risolvere sistemi di equazioni lineari di piccole o medie dimensioni.

Ecco alcuni esempi di metodi diretti per la risoluzione dei sistemi lineari:

1. **Sostituzione:** questo metodo consiste nel risolvere una delle equazioni del sistema per una delle variabili e sostituire questa soluzione nelle altre equazioni, riducendo gradualmente il sistema a un sistema di equazioni con meno variabili.
2. **Cramer:** questo metodo utilizza la regola di Cramer per calcolare la soluzione del sistema di equazioni. La regola di Cramer prevede la determinazione della soluzione del sistema attraverso il calcolo del determinante di una serie di matrici.
3. **Matrice inversa:** questo metodo consiste nel calcolare l'inversa della matrice dei coefficienti del sistema e moltiplicarla per il vettore dei termini noti per ottenere la soluzione del sistema.

4. **Gauss:** questo metodo, anche noto come eliminazione gaussiana, consiste nel ridurre il sistema di equazioni in una forma triangolare superiore mediante una serie di operazioni di eliminazione.
5. **LU:** questo metodo, anche noto come fattorizzazione LU, consiste nella decomposizione della matrice dei coefficienti in due matrici triangolari, una inferiore e una superiore, e nella successiva risoluzione di due sistemi triangolari.

I metodi diretti possono essere molto efficaci per la risoluzione di sistemi di equazioni lineari di dimensioni relativamente piccole o medie, ma diventano impraticabili per sistemi di grandi dimensioni, a causa della loro complessità computazionale elevata. In questi casi, sono generalmente preferiti i metodi iterativi.

METODI ITERATIVI

I metodi iterativi per la risoluzione dei sistemi lineari sono basati sulla ripetizione di un'operazione di aggiornamento della soluzione fino a quando si raggiunge una soluzione approssimata che soddisfi una certa tolleranza predefinita. Questi metodi sono solitamente utilizzati per risolvere sistemi di equazioni lineari di grandi dimensioni, in cui i metodi diretti sarebbero troppo costosi in termini di memoria e di tempo di calcolo.

Ecco alcuni esempi di metodi iterativi per la risoluzione dei sistemi lineari:

1. **Jacobi:** questo metodo prevede l'iterazione di una formula che aggiorna ogni componente del vettore soluzione con i valori delle componenti del vettore soluzione nella precedente iterazione.
2. **Gauss-Seidel:** questo metodo è simile al metodo di Jacobi, ma utilizza i valori delle componenti aggiornate del vettore soluzione durante l'iterazione, anziché i valori delle componenti della soluzione nella precedente iterazione.
3. **Gradiente coniugato:** questo metodo è un algoritmo di ricerca di una soluzione in uno spazio di vettori lineari e prevede l'iterazione di una formula che produce la direzione di ricerca ottima e la dimensione del passo ottimo.
4. **Richardson:** questo metodo è simile al metodo di Jacobi, ma prevede l'utilizzo di un parametro di rilassamento che controlla la velocità di convergenza dell'iterazione.
5. **Krylov:** questo metodo è basato sulla costruzione di una sequenza di vettori ortogonali, generati dalla moltiplicazione della matrice dei coefficienti del sistema per una serie di vettori iniziali.

I metodi iterativi sono generalmente meno costosi dei metodi diretti in termini di tempo di calcolo e di memoria richiesta, ma possono richiedere molte iterazioni per ottenere una soluzione approssimata con una tolleranza accettabile. La scelta del metodo iterativo migliore dipende dalle caratteristiche del sistema di equazioni lineari da risolvere e dalla precisione richiesta per la soluzione.

ALGORITMO GENERICO

L'algoritmo generico per la soluzione di sistemi lineari con un metodo iterativo consiste tipicamente nei seguenti passi:

1. **Inizializzazione:** Si parte da una soluzione iniziale, solitamente un vettore nullo o un vettore qualsiasi.
2. **Iterazione:** Si esegue un numero finito o infinito di iterazioni, aggiornando il vettore soluzione ad ogni iterazione.
3. **Test di convergenza:** Ad ogni iterazione, si verifica se la soluzione ottenuta è accettabile rispetto alla precisione richiesta. Se la soluzione è accettabile, l'algoritmo si ferma e restituisce la soluzione trovata.

4. Aggiornamento della soluzione: Per ogni iterazione, si calcola il nuovo vettore soluzione utilizzando una formula iterativa.
5. Verifica del criterio di arresto: Si verifica se il criterio di arresto è stato soddisfatto. Il criterio di arresto può essere un numero massimo di iterazioni, una tolleranza sulla norma del residuo, una tolleranza sulla differenza tra le soluzioni di due iterazioni successive, o un altro criterio di convergenza.
6. Ripetizione dell'iterazione: Se il criterio di arresto non è soddisfatto, si ripete il processo a partire dal passo 3, usando il nuovo vettore soluzione calcolato al passo 4.
7. Restituzione della soluzione: Alla fine delle iterazioni, si restituisce la soluzione approssimata trovata.

I passi specifici dell'algoritmo dipendono dal metodo iterativo utilizzato per risolvere il sistema di equazioni lineari. Ad esempio, il metodo di Jacobi prevede l'iterazione di una formula che aggiorna ogni componente del vettore soluzione, mentre il metodo del gradiente coniugato prevede l'iterazione di una formula che produce la direzione di ricerca ottima e la dimensione del passo ottimo.

JACOBI VS GAUSS-SEIDEL

Funzionamento:

- Jacobi: il metodo di Jacobi prevede di calcolare ciascuna componente del nuovo vettore soluzione come una funzione dei valori correnti delle altre componenti del vettore soluzione. In particolare, il valore della i -esima componente del nuovo vettore soluzione è calcolato come una funzione dei valori correnti delle altre componenti del vettore soluzione.
- Gauss-Seidel: il metodo di Gauss-Seidel è una variante del metodo di Jacobi, che prevede di utilizzare le componenti del nuovo vettore soluzione appena calcolate per calcolare le componenti successive. In particolare, il valore della i -esima componente del nuovo vettore soluzione è calcolato come una funzione del valore corrente della i -esima componente del vettore soluzione e dei valori correnti delle componenti successive.

Complessità:

- Jacobi: richiede un numero fisso di operazioni per ogni iterazione, indipendentemente dalle dimensioni del sistema lineare. Tuttavia, è noto per la sua lenta convergenza e per il fatto che richiede che la matrice dei coefficienti sia diagonalmente dominante per garantire la convergenza $O(n^2)$.
- Gauss-Seidel: richiede un numero variabile di operazioni per ogni iterazione, in base alle dimensioni del sistema lineare. Tuttavia, è noto per la sua rapida convergenza, in particolare se la matrice dei coefficienti è triangolare inferiore o se è diagonalmente dominante $O(n^2)$.

Formula per il calcolo delle iterazioni:

- Jacobi:

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left[b_i - \sum_{j=1, j \neq i}^n a_{ij} x_j^{(k)} \right] \quad \forall i = 1, 2, \dots, n$$

- Gauss-Seidel:

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left[b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^n a_{ij} x_j^{(k)} \right] \quad \forall i = 1, 2, \dots, n$$

Convergenza:

- Jacobi: il metodo di Jacobi converge se e solo se la matrice dei coefficienti è diagonalmente dominante. In generale, la convergenza del metodo di Jacobi è lenta e può richiedere un numero elevato di iterazioni per raggiungere la soluzione.
- Gauss-Seidel: il metodo di Gauss-Seidel converge se e solo se la matrice dei coefficienti è a dominanza diagonale. In generale, la convergenza del metodo di Gauss-Seidel è più rapida di quella del metodo di Jacobi e può raggiungere la soluzione in meno iterazioni. Inoltre, il metodo di Gauss-Seidel è convergente anche per alcune matrici dei coefficienti che non sono diagonalmente dominanti.

In sintesi, il metodo di Gauss-Seidel è generalmente preferito rispetto al metodo di Jacobi per la sua convergenza più rapida e per il fatto che può essere applicato anche a matrici dei coefficienti che non sono diagonalmente dominanti.

TEST DEI PROGRAMMI

Per eseguire i programmi relativi agli algoritmi di Jacobi e Seidel è necessario eseguire gli script:

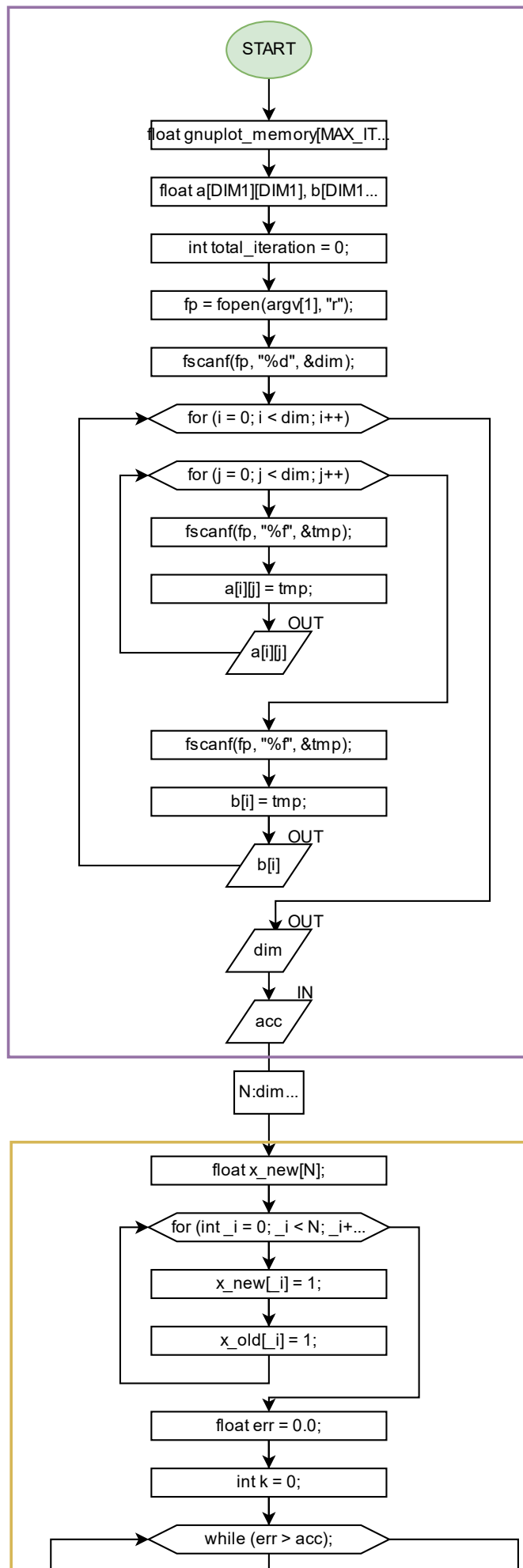
`./jacobi.sh [NOME_FILE_INPUT]`

`./seidel.sh [NOME_FILE_INPUT]`

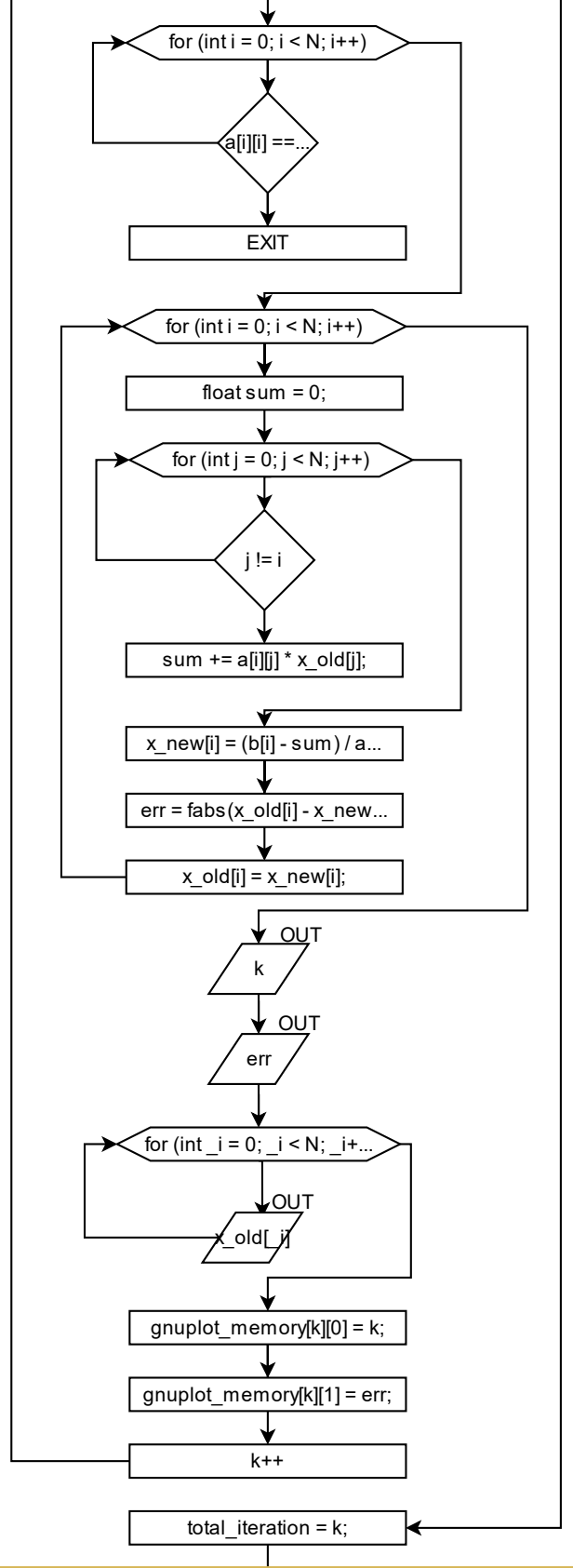
Si specifica che il file deve essere presente nella cartella **data/** e che se eseguiti tramite script è possibile generare anche i grafici degli errori nelle iterazioni

FLOWCHART JACOBI

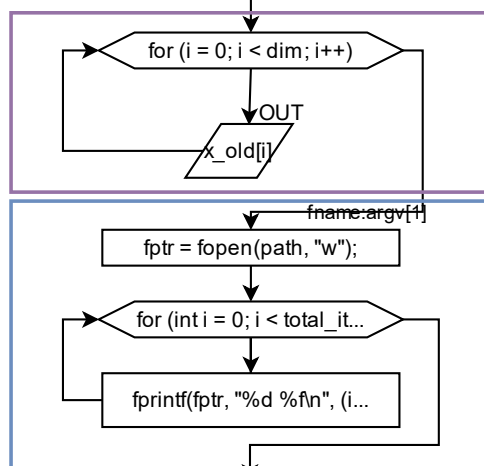
MAIN



JACOBI



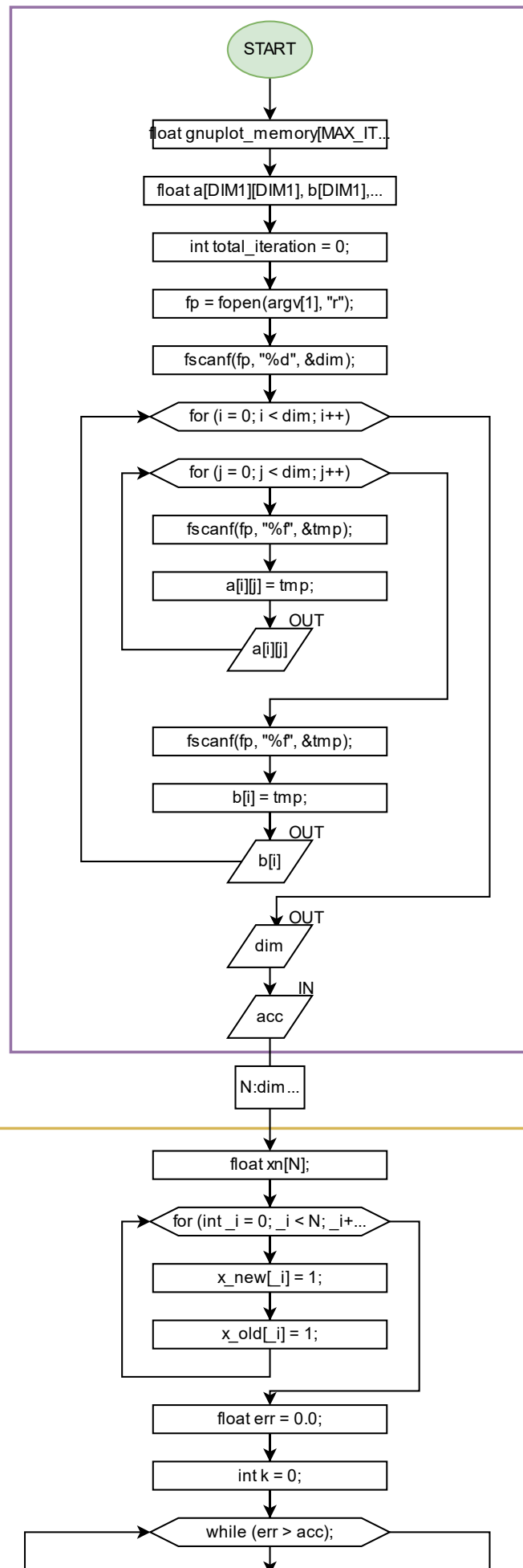
WRITE GNU PLOT



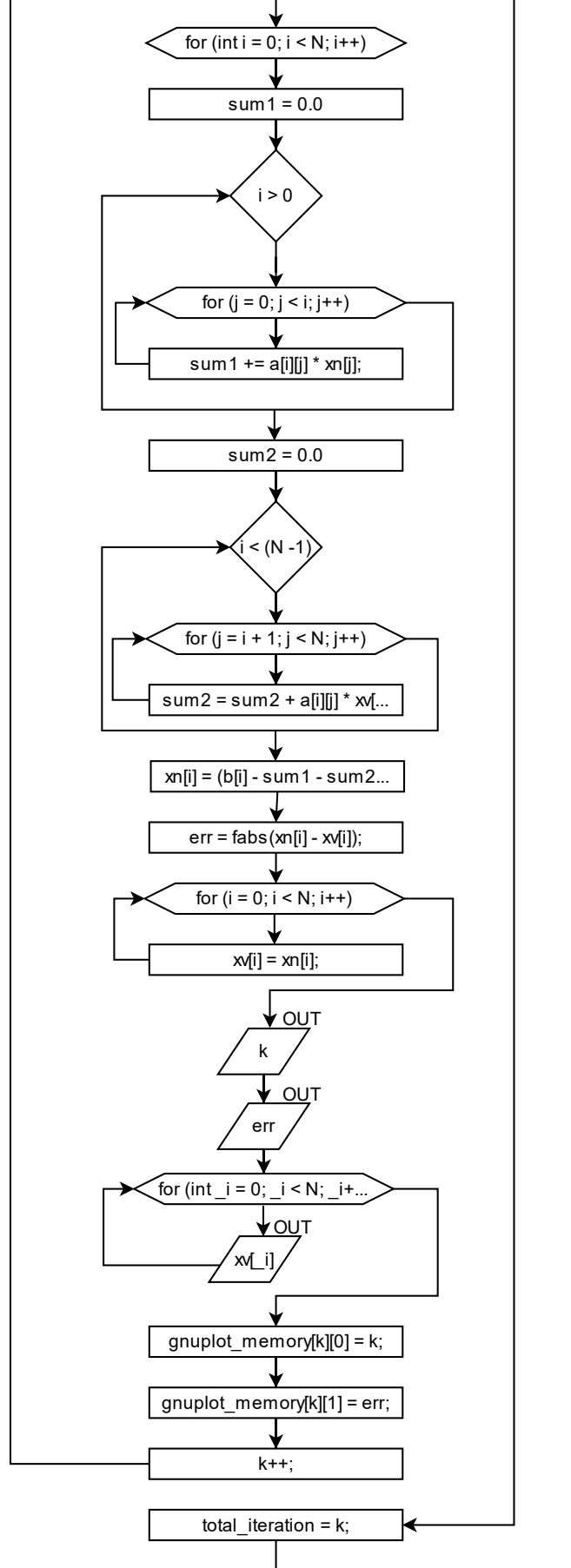
END

FLOWCHART SEIDEL

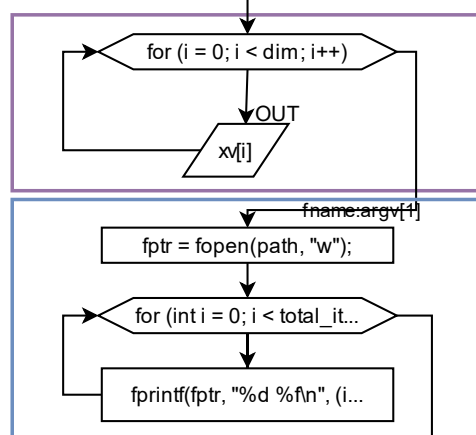
MAIN

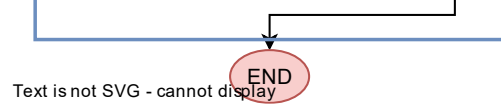


JACOBI



WRITE GNU PLOT





Numero di iterazioni necessario per raggiungere una data accuratezza

Accuratezza = 0.1

	ch10.2es3.txt	ch10.2es4.txt	ch10.2es23.txt	ch10.2es11.txt	ch10.2es12.txt
JACOBI	2	3	1	1	26
SEIDEL	3	4	3	3	26

Accuratezza = 0.01

	ch10.2es3.txt	ch10.2es4.txt	ch10.2es23.txt	ch10.2es11.txt	ch10.2es12.txt
JACOBI	5	5	2	1	26
SEIDEL	5	6	5	4	26