**LSSolver**

Metodi del Calcolo Scientifico

AA 2022-2023

Progetto 1.bis [Algebra lineare numerica]

Mini libreria per sistemi lineari

Lecchi Gabriele - 852134

Titta Lorenzo – 852107

**INTRODUZIONE**

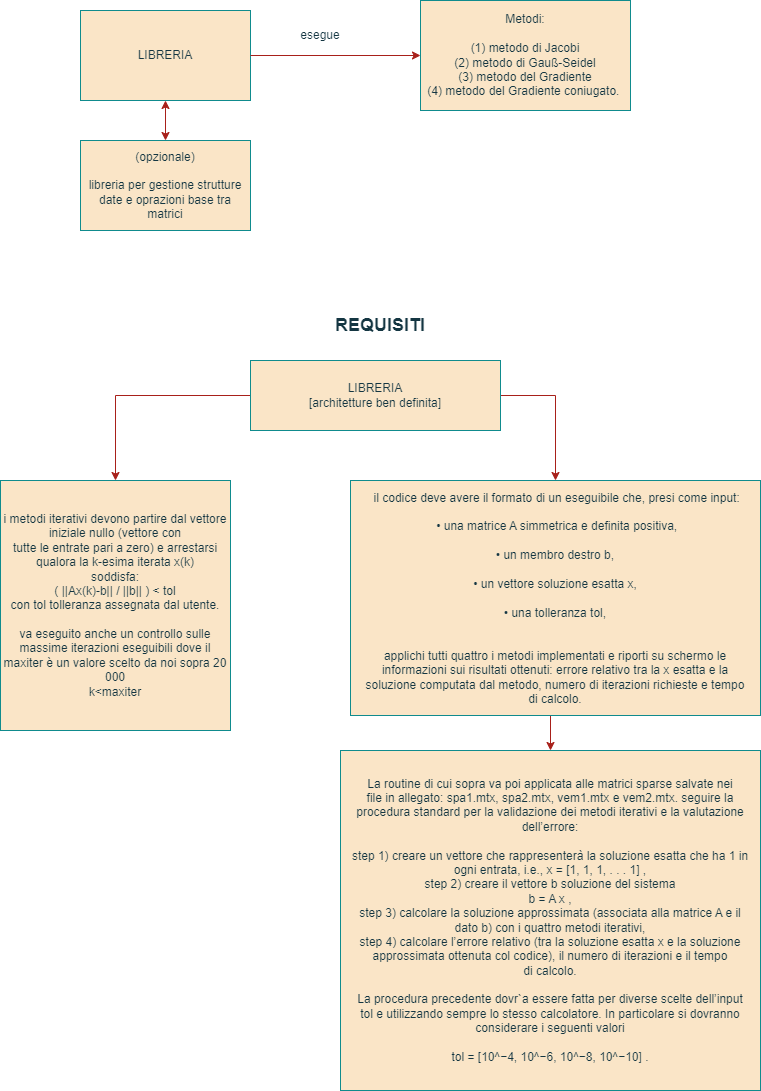
La richiesta di questo progetto prevedeva la creazione di una libreria con un linguaggio a nostra discrezione (nel nostro caso la scelta è ricaduta su Java) che riuscisse ad eseguire i 4 metodi iterativi qui menzionati limitandosi a matrici simmetriche e definite positive :

* Jacobi
* Gauß-Seidel
* Gradiente
* Gradiente coniugato

Per la gestione di: matrici, vettori e operazioni tra di essi ci siamo appoggiati a due librerie offerte da java, ovvero: Maths che è una delle librerie più utilizzate e la4j che per quanto riguarda le operazioni tra matrici e vettori risulta essere una delle più efficienti.

Tutti i metodi richiesti sono stati implementati totalmente da noi senza l’utilizzo di metodi già definiti da librerie di appoggio, di seguito verranno descritti e confrontati.

Il primo passaggio è stato quello di rappresentare i requisiti della richiesta in un grafico per mettere in chiaro i passaggi principali da svolgere, le task principali da implementare e come andare a progettare l’architettura, nella pagina successiva si può vedere il seguente grafico.



**ARCHITETTURA**

Per l’architettura abbiamo pensato di mantenere una struttura il più possibile semplice e pulita senza creare un solo file di pieno di codice.

Abbiamo diviso la libreria in 5 differenti file (una classe padre e quattro classi figlie):

La classe LSSolver è la classe padre dove vengono istanziate le varie matrici dai differenti file forniti da e-learning. Il principale compito di questa classe è fornire tutto ciò che serve alle classi figlie per l’esecuzione del metodo iterativo necessario per la risoluzione dei sistema lineare.

Le restanti 4 classi rappresentano invece i 4 metodi richiesti sotto la forma di classi figlie di LSSolver. In queste classi viene inizialmente istanziato un oggetto della loro classe padre tramite i loro costruttore, e successivamente da un file eseguibile esterno creato appositamente per testare la libreria viene chiamata la funzione che eseguirà effettivamente il metodo generato nella classe riportando tutto a console.

Di seguito un immagine che rappresenta la disposizione della libreria all’interno del package.

Immagine che contiene testo

Descrizione generata automaticamente

**JACOBI**