Parte 1

Escriba una rutina para resolver el sistema de ecuaciones lineales $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ con:

$$\boldsymbol{A} = \begin{pmatrix} 1 & 1 & 1 \\ 2 & 3 & 1 \\ 1 & -1 & -1 \end{pmatrix}, \qquad \boldsymbol{b} = \begin{pmatrix} 4 \\ 9 \\ -1 \end{pmatrix}$$

usando la descomposición LU y refinamiento iterativo.

La implementación de la descomposición LU se hizo a partir de un algoritmo de eliminación gausiana en donde se sobreescribe la matriz a descomponer, entendiendo que los elementos de la diagonal de la matriz inferior son igual a 1.

Luego, se definen implementaciones del método de substitución hacia atrás y hacia adelante, ambos muy similares. Cabe destacar que se aprovecha la estructura matricial para realizar la suma que está dentro del elemento x_i como un producto interno. Finalmente, el refinamiento iterativo implementado es ingenuo pero suficiente, puesto que la solución directa es ya muy precisa, inclusive con una tolerancia de $10e-6$ no actualiza el valor de la solución.

Se obtiene que $\boldsymbol{x} = (1.5,\ 1.75,\ 0.75)^\top$

Parte 2

Implementar el operador Sweep usando su lenguaje de programación favorito.

Nuevamente se hace uso de operaciones vectoriales con tal de acelerar el proceso y mejorar la legibilidad. Primero se pobla la matriz resultante con la fórmula general utilizando un producto externo, luego se sobreescriben la fila y columna asociada al operador (es decir, a la k-ésima fila y columna al operar Sweep(k)). Finalmente, se sobreescribe el k-ésimo elemento de la diagonal.

Parte 3

Pruebe su rutina para obtener la inversa de una matriz usando el operador Sweep con la siguiente matriz:

$$\boldsymbol{B} = \begin{pmatrix} 30 & 16 & 46 \\ 16 & 10 & 26 \\ 46 & 26 & 72 \end{pmatrix}$$

Con la implementación realizada se obtuvo:

$$\prod_{i=1}^{3} \mathrm{Sweep}(i)\boldsymbol{B} = \begin{pmatrix} 0.333 & 0.533 & 1.533 \\ -0.533 & 1.466 & 1.466 \\ 1.533 & 1.466 & 1.466 \end{pmatrix}$$

Lo interesante es que $\prod_{i=1}^{3} \mathrm{Sweep}(i)\boldsymbol{B}$ no es la inversa de $\boldsymbol{B}$ y se debe a que la matrix $\boldsymbol{B}$ es singular!

Parte 4

Verifique que $\boldsymbol{B}$ es matriz semidefinida positiva usando la factorización Cholesky. ¿En cuál etapa del algoritmo el procedimiento falla?

A pesar que el algoritmo implementado de la descomposición Cholesky no arroja ningún error numérico, se tiene que un elemento (el último) de la diagonal es nulo, no cumpliendo así que todos los elementos de la diagonal de $\boldsymbol{G}$ sean positivos (donde $\boldsymbol{G}$ cumpla $\boldsymbol{B} = \boldsymbol{G}^\top \boldsymbol{G}$).

La implementación arrojó que:

$$\boldsymbol{G} = \begin{pmatrix} 5.477 & 2.921 & 8.398 \\ 0 & 1.211 & 1.211 \\ 0 & 0 & 0 \end{pmatrix}$$

Sin embargo, note que igual se cumple $\boldsymbol{B} = \boldsymbol{G}^\top \boldsymbol{G}$. Probablemente sea un problema de estabilidad o precisión.