Aufgabe 1 (ca. 45 Min.):

Gegeben ist das lineare Gleichungssystem $\boldsymbol{Ax} = \boldsymbol{b}$ mit

$$\boldsymbol{A} = \begin{pmatrix} 1 & 0 & 2 \\ 0 & 1 & 0 \\ 10^{-4} & 0 & 10^{-4} \end{pmatrix} \text{ und } \boldsymbol{b} = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}$$

a) Bestimmen Sie die Kondition von $\boldsymbol{A}$ bzgl. der $\infty-$Norm. Sie dürfen $\boldsymbol{A}^{-1}$ mit Python berechnen.

b) Für $\varepsilon > 0$ sei die fehlerbehaftete rechte Seite

$$\tilde{b} = \begin{pmatrix} 1 \\ 1 \\ \varepsilon \end{pmatrix}$$

gegeben. Wie gross darf ε maximal sein, wenn die Abschätzung des relativen Fehlers $\frac{\|\tilde{x}-x\|_\infty}{\|x\|_\infty}$ der Lösung $\tilde{x}$ mit Hilfe der Kondition aus Aufgabe a) höchstens 1% sein darf?

c) Welcher relative Fehler $\frac{\|\tilde{x}-x\|_\infty}{\|x\|_\infty}$ ergibt sich für die Lösung $\tilde{x}$ mit der fehlerbehafteten rechten Seite aus Aufgabe b) und dem dort berechneten maximalen ε tatsächlich?

d) Gehen Sie nun davon aus, dass nun zusätzlich noch jede Komponente von $\boldsymbol{A}$ um bis $1e-7$ gestört sein kann. Wiederholen Sie mit dieser zusätzlichen Information die Berechnung aus b).

Hinweis: Matrixinverse mit Python bilden
Gleichungssysteme mit Python lösen

a) $$\text{cond}(A) = \|A\|_\infty \cdot \|A^{-1}\|_\infty = \left\| \begin{pmatrix} 1 & 0 & 2 \\ 0 & 1 & 0 \\ 10^{-4} & 0 & 10^{-4} \end{pmatrix} \right\| \cdot \left\| \begin{pmatrix} -1 & 0 & 20'000 \\ 0 & 1 & 0 \\ 1 & 0 & -10'000 \end{pmatrix} \right\|$$

$$= 3 \cdot 20'001 = 60'003$$

b) $\|b\|_\infty = 1$

$\|b - \tilde{b}\|_\infty = |\varepsilon|$

$$\frac{\|x-\tilde{x}\|_\infty}{\|x\|_\infty} \leq \text{cond}(A) \cdot \frac{\|b-\tilde{b}\|_\infty}{\|b\|_\infty}$$

$$\leq 60003 \cdot \frac{|\varepsilon|}{1}$$

$\hookrightarrow \quad 60003\, \varepsilon \leq 0.01$

$$\varepsilon \leq \frac{0.01}{60003}$$

$$\varepsilon \leq 1.66658 \cdot 10^{-7}$$

Ergebnis $= 1.67 \cdot 10^{-7}$

c) $x = A \backslash b = \begin{pmatrix} -1 \\ 1 \\ 1 \end{pmatrix}$ // via np.linalg.solve

$\tilde{x} = A \backslash \tilde{b} = \begin{pmatrix} -0.943 \\ 1.0 \\ 0.971 \end{pmatrix}$ // via np.linalg.solve

```
"C:\Users\armandoshala\
[[-1.]
 [ 1.]
 [ 1.]]
[[-0.94399242]
 [ 1.        ]
 [ 0.97199621]]
```

$$\frac{\|x - \tilde{x}\|_\infty}{\|x\|_\infty} = 3.33317e-3$$

d) $\|A - \tilde{A}\|_\infty = 3e-7$

$\|A\|_\infty = 3$

$$\frac{\|A - \tilde{A}\|_\infty}{\|A\|_\infty} = \frac{3e-7}{3} = 1e-7$$

$$\operatorname{cond}(A) \cdot \frac{\|A - \tilde{A}\|_\infty}{\|A\|_\infty} = 60'003 \cdot 10^{-7} = 0.60003 \cdot 10^{-2} < 1$$

$$\frac{\|x - \tilde{x}\|_\infty}{\|x\|_\infty} \leq \frac{\operatorname{cond}(A)}{1 - \operatorname{cond}(A) \cdot \frac{\|A - \tilde{A}\|_\infty}{\|A\|_\infty}} \cdot \left(\frac{\|A - \tilde{A}\|_\infty}{\|A\|_\infty} \quad \frac{\|b - \tilde{b}\|_\infty}{\|b\|_\infty} \right)$$

$$\leq \frac{60003}{1 - 0.60003 \cdot 10^{-2}} \left(10^{-7} + \varepsilon \right) \leq 0.01$$

$$\Rightarrow \varepsilon \leq 6.56583 \cdot 10^{-8}$$

Aufgabe 2 (ca. 45 Min.): Aufg. 1 als Python zur Automatisierung // Wichtig für SEP

Schreiben Sie ein Funktion $[x, \widetilde{x}, dx_{max}, dx_{obs}]$ = `Name_S9_Aufg2`$[A, \widetilde{A}, b, \widetilde{b}]$:

- Input: Matrix A und Vektor b des linearen Gleichungssystems $Ax = b$, sowie die gestörte Matrix $\widetilde{A}$ und Vektor $\widetilde{b}$ des gestörten Gleichungssystems $\widetilde{A}\widetilde{x} = \widetilde{b}$.
- Output: Lösung x des linearen Gleichungssystems $Ax = b$ und Lösung $\widetilde{x}$ des gestörten Gleichungssystems $\widetilde{A}\widetilde{x} = \widetilde{b}$. Ausserdem die obere Schranke des relativen Fehlers dx_{max} von x gemäss Skript, also $dx_{max} = \frac{\mathrm{cond}(A)}{1-\mathrm{cond}(A)\cdot\frac{\|A-\widetilde{A}\|}{\|A\|}} \cdot \left(\frac{\|A-\widetilde{A}\|}{\|A\|} + \frac{\|b-\widetilde{b}\|}{\|b\|}\right)$ in der Unendlich-Norm, und der tatsächliche relative Fehler $dx_{obs} = \frac{\|x-\widetilde{x}\|_\infty}{\|x\|_\infty}$. Falls die Bedingung $\mathrm{cond}(A) \cdot \frac{\|A-\widetilde{A}\|}{\|A\|} < 1$ für die Berechnung von dx_{max} nicht erfüllt ist, soll für dx_{max} der Wert 'NaN' (Not a Number) ausgegeben werden.
- Überprüfen Sie mit Ihrer Funktion für sich die Resultate von Aufgabe 3 der Serie 8.

Tipp: Sie dürfen Python-Funktionen und Operatoren verwenden, z.B. `np.linalg.solve()` für die Lösung der linearen Gleichungssysteme oder die Funktionen `np.linalg.cond(A,np.inf)` resp. `np.linalg.norm(b,np.inf)` für die Berechnung der Kondition resp. der Norm (für Details siehe die Beschreibung dieser Funktionen im Numpy-Manual).

Aufgabe 3 (ca. 30 Minuten): "Realitätscheck"

Testen Sie, inwiefern dx_{max} aus Aufgabe 2 eine realistische obere Schranke für dx_{obs} ist. Schreiben Sie dazu ein Skript `Name_S9_Aufg3.py` und gehen Sie folgendermassen vor:

- Defnieren Sie eine for-Schleife mit 1000 Iterationen. Erzeugen Sie für jede Iteration mittels der Python-Funktion `np.random.rand()` eine zufällige 100×100 Matrix $\boldsymbol{A}$ und einen zufälligen $100\times$ Vektor $\boldsymbol{b}$ (lesen Sie die Eigenschaften von `rand()` nach). Erzeugen Sie zusätzlich für jede Iteration eine gestörte Matrix $\widetilde{\boldsymbol{A}} = \boldsymbol{A} + \mathrm{rand}(100, 100)/10^5$ und einen gestörten Vektor $\widetilde{\boldsymbol{b}} = \boldsymbol{b} + \mathrm{rand}(100, 1)/10^5$
- Berechnen Sie für jede Iteration mit Ihrer Funktion aus Aufgabe 2 dx_{max}und dx_{obs}. Schreiben Sie dx_{max}, dx_{obs} sowie das Verhältnis dx_{max}/dx_{obs}in Vektoren und stellen Sie diese mit `plt.semilogy()` grafisch dar.
- Schreiben Sie Ihren Kommentar, ob dx_{max} in dieser Versuchsanordnung eine realistische obere Schranke für dx_{obs} ist, in Ihr Skript.