You will write a function

- \( x = \text{ge}\_\text{precision}(A, b, \text{precision}) \);
- \( x = \text{ge}\_\text{pp}\_\text{precision}(A, b, \text{precision}) \);

The function \( \text{ge}\_\text{precision} \) should use Gaussian elimination with no pivoting to solve the system of equations

\[
Ax = b.
\]

Each arithmetic operation should be done with simulated floating point arithmetic with precision bits using \( \text{my}\_\text{plus} \) and \( \text{my}\_\text{mult} \). To compute the division \( a/b \) with a specified precision you may “cheat” and use \( \text{my}\_\text{mult}(a, 1/b, \text{precision}) \). The function \( \text{ge}\_\text{pp}\_\text{precision} \) should use partial pivoting. I will post code for Gaussian elimination and Gaussian elimination with partial pivoting that you may use as scaffolding; but remember to change every arithmetic operation to one with simulated precision. I suggest that you write helper functions

- \( xpy = \text{add}\_\text{arrays}\_\text{precision}(x, y, \text{precision}) \)
- \( ax = \text{scale}\_\text{array}\_\text{precision}(a, x, \text{precision}) \)

to clean up your code, but this is not necessary if you do not want.