You will write a function

\[ [Q \ R] = \text{my\_gs\_qr}(A, \text{type}, \text{precision}); \]

The function \textit{my\_gs\_qr} should take as inputs an \( m \times n \) real matrix \( A \), with \( m \geq n \), a number type, with type equal to 1 or 0, and a number precision, where precision is an integer between 0 and 9, inclusive. The function should return an \( m \times n \) matrix \( Q \), and the upper triangular \( n \times n \) matrix \( R \), computed with simulated floating point arithmetic with precision base ten bits (and matlabs internal full precision arithmetic if \( \text{precision} = 0 \)). If type is 1, the code should use modified Gram-Schmidt, and if type is 0, the code should use classical Gram-Schmidt. All arithmetic operations used in all the computations should be implemented using simulated precision. Use the pseudocode in exercise 5.5.10 of MAALA.