Gauß-Algorithmus mit Spaltenpivotisierung

Eingabe: Matrix $A = (a_{i,j})_{i,j=1}^n$, rechte Seite $b = (b_j)_{j=1}^n$
Ausgabe: Lösung $x = (x_j)_{j=1}^n$

Sei $p = (p_j)_{j=1}^n$ ein Vektor zur Speicherung der Zeilenvertauschungen

Initialisiere: $p_j = j$, $j = 1, \ldots, n$

// Berechnung der LR-Zerlegung und Auflösen von $Ly = b$

für $k = 1, 2$ bis $n - 1$

  bestimme Index m, so dass $|a_{p_m,k}| = \max\limits_{k \le i \le n} |a_{p_i,k}|$

  falls $|a_{p_m,k}| > |a_{p_k,k}|$

    // tausche implizit Zeile k und Zeile m:

    setze $tmp = p_k$, $p_k = p_m$, $p_m = tmp$

  für $i = k + 1, k + 2$ bis n

    $a_{p_i,k} = a_{p_i,k} / a_{p_k,k}$

    für $j = k + 1, k + 2$ bis n

      $a_{p_i,j} = a_{p_i,j} - a_{p_i,k} a_{p_k,j}$

    $b_{p_i} = b_{p_i} - a_{p_i,k} b_{p_k}$

// Löse $Lz = y$

für $i = n, n - 1$ bis 1

  $z_{p_i} = (b_{p_i} - \sum_{k>i} a_{p_i,k} z_{p_k}) / a_{p_i,i}$

// Rücktausch

für $j = 1, 2$ bis n

  $x_i = z_{p_i}$

gib x zurück