Algorithme de GAUSS

```
Procedure GAUSS (A : Matrice, b : Vecteur, var X : Vecteur) ;
/* Objectifs : Resoudre le systeme (S) :A.X = b par la methode de Gauss
            avec strategie du pivot partiel. */
/* Entrees : − A ∈ M_n(ℝ) : matrice du systeme, inversible ;
            − b ∈ ℝ^n : vecteur 2nd membre ; */

/* Sorties : X ∈ ℝ : vecteur solution du systeme ; */

Debut
    Elimination(A, b) ;
    Remontee(A, b, X) ;
    Renvoyer(X) ;
STOP
```

```
Procedure Elimination (var A : Matrice, var b : Vecteur) ;
/* Objectifs : Rendre le systeme, (S) : A.X = b, sup−triangulaire
            par la methode d'elimination de Gauss avec strategie du pivot partiel */
Variables : k, i_max : Entier ; ℓ_ik : Reel ;

Debut
    /* On fait l'elimination sur chaque colonne k = 1, . . . , n − 1. */
    Pour k = 1(1)n − 1 Faire
        /* Pour l'elimination sur la colonne k, on commence par determiner le pivot
        par la methode du pivot partiel */

        pivot_partiel(A, b, k) ;

        /* l'element A[k, k] est donc le pivot pour cette elimination */
        /* Procedons alors a l'elimination proprement dite */
        Pour i = k + 1(1)n Faire
            ℓ_ik ←− A[i, k] / A[k, k] ;
            b[i] ←− b[i] − ℓ_ik*b[k] ;
            Pour j = k + 1(1)n Faire
                A[i, j] ←− A[i, j] − ℓ_ik*A[k, j] ;
            finPour
        finPour
    finPour
    Renvoyer(A, b) ;
STOP
```

<u>Fonction</u> pivot_partiel (**<u>Var</u>** **A** : **<u>Matrice</u>**, **<u>Var</u>** b : **<u>Vecteur</u>**, k : **<u>Entier</u>**) ;
/* Objectifs : Trouver l'indice i_{max} du pivot partiel sur la colonne k de la matrice A
donne par : $i_{max} = \underset{k \leqslant i \leqslant n}{\mathbf{argmax}} |\mathbf{A}[i,k]|$.
Ensuite permuter si eventuellement les lignes k et i_{max} */

<u>Variables</u> : i, j, i_{max} : **<u>Entier</u>** ; **aux** : **<u>Reel</u>** ;
<u>Debut</u>
$i_{max} \longleftarrow k$;
<u>Pour</u> $i = k+1(1)n$ **<u>Faire</u>**
<u>Si</u> $\mathbf{A}[i,k] > \mathbf{A}[i_{max},k]$ **<u>Alors</u>**
$i_{max} \longleftarrow i$;
<u>finSi</u>
<u>finPour</u>
<u>Si</u> $i_{max} \neq k$ **<u>Alors</u>**
/* On permute les lignes k et i_{max} */

aux $\longleftarrow$ $\mathbf{b}[k]$; $\mathbf{b}[k] \longleftarrow \mathbf{b}[i_{max}]$; $\mathbf{b}[i_{max}] \longleftarrow$ **aux** ;
<u>Pour</u> $j = k(1)n$ **<u>Faire</u>**
aux $\longleftarrow \mathbf{A}[k,j]$; $\mathbf{A}[k,j] \longleftarrow \mathbf{A}[i_{max},j]$; $\mathbf{A}[i_{max},j] \longleftarrow$ **aux** ;
<u>finPour</u>

<u>finSi</u>
Renvoyer(A, b) ;

<u>STOP</u>

procedure Remontee(A : **<u>Matrice</u>**, **b** : **<u>Vecteur</u>**, **var X** : **<u>Vecteur</u>**) ;
/* Objectifs : la matrice A du systeme, (S) : A.X = b, a resoudre est deja sup−triangulaire.
Il ne reste qu'a tirer les valeurs des inconnues en remontant les
equations du systeme partant de la derniere a la premiere */
<u>Variables</u> : i, j : **<u>Entier</u>** ;
<u>Debut</u>
/* On commence par la derniere inconnue x_n */
$\mathbf{X}[n] \longleftarrow \mathbf{b}[n]$ / $\mathbf{A}[n,n]$;

/* On deduit le reste d'inconnues $x_{n-1}, x_{n-2}, \ldots, x_1$ */
<u>Pour</u> $i = n-1\ (-1)\ 1$ **<u>Faire</u>**
$\mathbf{X}[i] = \mathbf{b}[i]$;
<u>Pour</u> $j = i+1\ (1)\ n$ **<u>Faire</u>**
$\mathbf{X}[i] \longleftarrow \mathbf{X}[i] - \mathbf{A}[i,j] * \mathbf{X}[j]$;
<u>finPour</u>
$\mathbf{X}[i] \longleftarrow \mathbf{X}[i]$ / $\mathbf{A}[i,i]$;
<u>finPour</u>
Renvoyer(X) ;
<u>STOP</u>