TAB. 1: *Algorithme d'élimination de Gauss.*

entrées : a_{ij}, $1 \le i, j \le N$ et b_j, $1 \le j \le N$
représentant la matrice A et le second membre $\vec{b}$
du système linéaire originel
sorties : a_{ij}, $1 \le i < j \le N$ et b_j, $1 \le j \le N$
représentant la partie surdiagonale
du système triangulaire supérieur
et le nouveau second membre $\vec{b}$

Algorithme	Commentaires
Faire $i = 1$ à $N - 1$	Elimination de l'inconnue x_i
$p := 1/a_{ii}$	Inverse du i-ième pivot
Faire $j = i + 1$ à N $a_{ij} := p * a_{ij}$	Division de la i-ième ligne par le i-ième pivot (termes surdiagonaux uniquement)
$b_i := p * b_i$	Division de b_i par le i-ième pivot
Faire $k = i + 1$ à N	Elimination dans la k-ième équation
Faire $j = i + 1$ à N $a_{kj} := a_{kj} - a_{ki} * a_{ij}$	Soustraction de a_{ki} fois la nouvelle i-ième ligne à la k-ième ligne
$b_k := b_k - a_{ki} * b_i$	Soustraction de a_{ki} fois b_i à b_k
$p := 1/a_{NN}$	Inverse du N-ième pivot
$b_N := p * b_N$	Division de b_N par le N-ième pivot

TAB. 2: *Algorithme du choix du plus grand pivot.*

Choix du plus grand pivot en valeur absolue (à intercaler entre les deux premières lignes de l'algorithme d'élimination de Gauss)	
Algorithme	**Commentaires**
$k := i$	On élimine a priori l'inconnue x_i et donc i est fixé
$m := abs(a_{ii})$	On initialise $m =\mid a_{ii} \mid$
Faire $j = i + 1$ à N	Recherche du plus grand pivot en valeur absolue dans la i-ième colonne
$s := abs(a_{ji})$ Si $m < s$ alors $k := j$ et $m := s$	Après cette séquence, m est le plus grand pivot en valeur absolue et se trouve à la ligne k. (si $m = 0$ alors le système est singulier)
Si $k = i$ on saute ce qui suit	Si $k = i$ on n'a pas à modifier l'élimination de l'inconnue x_i dans l'algorithme d'élimination de Gauss
Faire $j = i$ à N	Echange des lignes k et i
$t := a_{ij}$ $a_{ij} := a_{kj}$ $a_{kj} := t$	Echange de a_{ij} et a_{kj}
$t := b_i$ $b_i := b_k$ $b_k := t$	Echange de b_i et b_k

Tab. 3: *Algorithme de décomposition LU.*

entrées : a_{ij}, $1 \leq i, j \leq N$ sont les coefficients de la matrice A	
sorties : a_{ij}, $1 \leq j \leq i \leq N$ sont les coefficients de la matrice L	
et a_{ij}, $1 \leq i < j < N$ sont les coefficients surdiagonaux de la matrice U	
Algorithme	**Commentaires**
Faire $i = 2$ à N $a_{1i} := a_{1i}/a_{11}$	Construction de la première ligne de U (la $1^{\text{ère}}$ col. de L est la $1^{\text{ère}}$ col. de A)
Faire $k = 2$ à $N - 1$	Colonnes de L et lignes de U
$a_{kk} := a_{kk} - \sum_{j=1}^{k-1} a_{kj} * a_{jk}$	Construction du pivot ℓ_{kk}
Faire $i = k + 1$ à N	
$a_{ik} := a_{ik} - \sum_{j=1}^{k-1} a_{ij} * a_{jk}$	Construction de la k-ième colonne de L
$a_{ki} := \dfrac{1}{a_{kk}} \left(a_{ki} - \sum_{j=1}^{k-1} a_{kj} * a_{ji} \right)$	Construction de la k-ième ligne de U
$a_{NN} := a_{NN} - \sum_{j=1}^{N-1} a_{Nj} * a_{jN}$	Construction du pivot ℓ_{NN}

TAB. 4: *Algorithme de Cholesky.*

entrées : $(a_{ij})_{1\leq j\leq i\leq N}$ représentant la partie triangulaire inférieure de A; (A est symétrique définie positive) **sorties** : $(a_{ij})_{1\leq j\leq i\leq N}$ représentant L qui satisfait $A = LL^T$	
Algorithme	**Commentaires**
$a_{11} := \sqrt{a_{11}}$	Construction de ℓ_{11}
Faire $i = 2$ à N $a_{i1} := a_{i1}/a_{11}$	Construction de la première colonne de L
Faire $k = 2$ à $N-1$	Parcours des colonnes de L
$a_{kk} := \left(a_{kk} - \sum_{j=1}^{k-1} a_{kj}^2\right)^{1/2}$	Construction de ℓ_{kk}
Faire $i = k+1$ à N $a_{ik} := \dfrac{1}{a_{kk}}\left(a_{ik} - \sum_{j=1}^{k-1} a_{ij} * a_{kj}\right)$	Construction de la k-ième col. de L
$a_{NN} := \left(a_{NN} - \sum_{j=1}^{N-1} a_{Nj}^2\right)^{1/2}$	Construction de ℓ_{NN}