

Pivot de Gauss

Objectif : inverser une matrice par la méthode du pivot.

Données :

Une matrice A de type array(1..n,1..n)

La taille n de la matrice

Description de l'algorithme

Algorithme itératif. A la $j^{\text{ème}}$ itération : on cherche sur la $j^{\text{ème}}$ ligne un terme non nul $a(j,k)$ avec $j \leq k \leq n$.

Si ce n'est pas possible, alors la matrice n'est pas inversible.

Sinon, on échange la colonne j et la colonne k, on multiplie la colonne j par $1/a(j,j)$, puis on applique l'opération $C[k] \leftarrow C[k] - a(j,k) * C[j]$ pour tout $k \neq j$. On applique en parallèle les mêmes opérations à la matrice unité, le résultat final est l'inverse de A.

Valeur de sortie

L'inverse de la matrice A

Implémentation MAPLE

```
> gauss:=proc(A) global n ;local M,In,R,i,j,k,m,temp,pivot;
# répliques de A et de la matrice unité :
M:=array(1..n,1..n); for i to n do for j to n do
M[i,j]:=A[i,j] od od;
R:=array(1..n,1..n); for i to n do for j to n do R[i,j]:=0 od;
R[i,i]:=1 od;
# boucle sur les colonnes :
for j to n do
# recherche d'un pivot sur la j-ème ligne :
k:=j; while(k<=n and M[j,k]=0) do k:=k+1 od; #ordre des tests
importants!
if k=n+1 then ERROR(`matrice non inversible`) #cas singulier
else
# échange colonne k et colonne j
for i from 1 to n do
temp:=M[i,j] ; M[i,j]:=M[i,k] ; M[i,k]:=temp;
temp:=R[i,j] ; R[i,j]:=R[i,k] ; R[i,k]:=temp od;
fi ;
pivot:=M[j,j];
# simplification de la j-ème colonne :
for i from 1 to n do M[i,j]:=M[i,j]/pivot;
R[i,j]:=R[i,j]/pivot od;
# nettoyage de la j-ème ligne :
for m from 1 to n do
if m<> j then temp:=M[j,m];
for i from 1 to n do
```

```

M[i,m]:=M[i,m]-temp*M[i,j];
R[i,m]:=R[i,m]-temp*R[i,j] od;
fi;
od;
od; #fin de la boucle j
eval(R) #résultat final
end proc;

```

gauss := proc(A)

local M, In, R, i, j, k, m, temp, pivot;

global n;

M := array(1 .. n, 1 .. n);

for i **to** n **do** **for** j **to** n **do** M[i,j] := A[i,j] **end do** **end do**;

R := array(1 .. n, 1 .. n);

for i **to** n **do** **for** j **to** n **do** R[i,j] := 0 **end do** R[i,i] := 1 **end do**;

for j **to** n **do**

k := j;

while k ≤ n **and** M[j,k] = 0 **do** k := k + 1 **end do**;

if k = n + 1 **then** ERROR(`matrice non inversible`)

else **for** i **to** n **do**

temp := M[i,j];

M[i,j] := M[i,k];

M[i,k] := temp;

temp := R[i,j];

R[i,j] := R[i,k];

R[i,k] := temp

end do

end if;

pivot := M[j,j];

for i **to** n **do** M[i,j] := M[i,j]/pivot; R[i,j] := R[i,j]/pivot **end do**;

for m **to** n **do**

if m ≠ j **then**

temp := M[j,m];

for i **to** n **do**

M[i,m] := M[i,m] - temp×M[i,j];

R[i,m] := R[i,m] - temp×R[i,j]

end do

end if

end do

end do;

eval(R)

end proc

Essais

> **n:=2; B:=array(1..2,1..2,[[1,2],[3,4]]);**

$n := 2$

$$B := \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$$

> **gauss(B) ;**

$$\begin{bmatrix} -2 & 1 \\ \frac{3}{2} & \frac{-1}{2} \end{bmatrix}$$

> **B:=array(1..2,1..2,[[1,2],[2,4]]) ;**

$$B := \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix}$$

> **n:=2;gauss(B) ;**

$n := 2$

Error, (in gauss) matrice non inversible