

```

1 restart;maple_mode(1);cas_setup(0,0,0,1,0,1e-10,10,[1,50,0,25],0,0,0);#radians,pas de cmplx, pas de Sqrt
Warning: some commands like subs might change arguments order , 0, 0, 0, 1, 0, 1e-10, 10, [1 50 0 25 ], 0, 0,
2 gcd(6,3);#Attention, a priori pour lui c'est des polynomes, on a dela chance il les normalise bien.
3
4
5
6
7
8
9
10
11
12

```

3

x

2

(2. 6)

1

1

[1 1]

// Success

$$\left(\begin{array}{cccc|cccc} 3 & 0 & 0 & 0 & 3 & 2 & -1 & -2 \\ 0 & 6 & 0 & 0 & 0 & 2 & 1 & 3 \\ 0 & 0 & 18 & 0 & 1 & 2 & -3 & -1 \\ 0 & 0 & 0 & 36 & 2 & 1 & 1 & -3 \end{array} , 49 \right)$$

// Success

$$\left(\begin{array}{cccc|cccc|cccc} 1 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 1 & 34 \\ 10 & -28 & 1 & 7 & 0 & 6 & 0 & 0 & 0 & -111 & 9315 & 316487 \\ -6444 & -987 & 717 & 703 & 0 & 0 & 18 & 0 & -1 & -220 & 18465 & 627368 \\ 18588 & 2844 & -2068 & -2027 & 0 & 0 & 0 & 1764 & 0 & -1 & 84 & 2854 \end{array} \right)$$

Prog	Edit	Ajouter	nxt	OK	Save
------	------	---------	-----	----	------

```

minval:=proc(A)
local m,u,v,i,j;
m:=0;u:=0;v:=0;
# NB: On retourne 0,0 si A est nulle
for i from 1 to dim(A)[1] do
for j from 1 to dim(A)[2] do
if ((abs(A[i,j])>0) and ((m=0) or (abs(A[i,j])<m))) then m:=abs(A[i,j]);u:=i;v:=j; end if;
od;
od;
u,v;end proc;
// Success
// End defining minval

proc(A)
local m,u,v,i,j;
m:=0;
u:=0;
v:=0;
for i from 1 to (dim(A))[1]+1/2 do
for j from 1 to (dim(A))[2]+1/2 do
if ((abs(A[i,j])>0) and ((m=0) or (abs(A[i,j])<m))) then
m:=abs(A[i,j]);
u:=i;
v:=j;
fi;
od;;
od;;
u,v;
end;

```

13 A:=matrix([6, 0, 7, 9], [6, 8, 6, 9], [3, 2, 4, 6], [3, 2, 3, 3]);

$$\begin{pmatrix} 6 & 0 & 7 & 9 \\ 6 & 8 & 6 & 9 \\ 3 & 2 & 4 & 6 \\ 3 & 2 & 3 & 3 \end{pmatrix}$$

14 minval(A);

(3, 2)

15 (i,j):=minval(A);

(3, 2)

16 U:=identity(4);

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

17 normalement il faudrait faire bouger l dans {1..4} sauf j, c'est plus simple de ne pas mettre de test, et de corriger U[j,j] ensuite

18

undef

19 for l from 1 to 4 do U[j,l]:=-iquo(A[i,l],A[i,j]) od:U[j,j]:=1:U;

(Done, Done, $\begin{pmatrix} 1 & 0 & 0 & 0 \\ -1 & 1 & -2 & -3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$)

20 A*U;

$$\begin{pmatrix} 6 & 0 & 7 & 9 \\ -2 & 8 & -10 & -15 \\ 1 & 2 & 0 & 0 \\ 1 & 2 & -1 & -3 \end{pmatrix}$$

21

22 Prog Edit Ajouter nxt OK Save

```
trans:=proc(A,i,j)
local n,U,V,l;
n:=dim(A)[1];
U:=identity(n);
V:=identity(n);
for l from 1 to n do
V[l,i]:=-iquo(A[l,j],A[i,j])
od;
//on corrige
V[i,i]:=1;
for l from 1 to n do
U[j,l]:=-iquo(A[i,l],A[i,j])
od;
//on corrige
U[j,j]:=1;
V*A*U;
end proc;
```

// Success
// End defining trans

```
proc(A,i,j)
local n,U,V,l;
n:=(dim(A))[1];
U:=identity(n);
V:=identity(n);
for l from 1 to n+1/2 do
V[l,i]:=-iquo(A[l,j],A[i,j]);
od;
V[i,i]:=1;
for l from 1 to n+1/2 do
U[j,l]:=-iquo(A[i,l],A[i,j]);
od;
```

V^A*U;

end;

23 A:=matrix([[6, 1, 7, 9], [6, 8, 6, 9], [3, 1, 4, 6], [3, 2, 1, 3]]);

6	1	7	9
6	8	6	9
3	1	4	6
3	2	1	3

24 trans(A,3,1);

0	-1	-1	-3
0	6	-2	-3
3	1	1	0
0	1	-3	-3

25

26 Prog Edit Ajouter

nxt

OK

Save

```
Zequiv:=proc(A)
k:=0;
n:=dim(A)[1];
l:=seq(0,k=1..n)
B:=A;
(i,j):=minval(B)
//minval retourne les coordonnees du minimum non nu.
//ou un couple impossible (Ex ici (0,0) en mode maple) si B est null
while((i,j)<>[0,0]) and size(B)>1 do
// attention, ne pas oublier de declarer les variables locales
// dans trans pour eviter les melange
B:=trans(B,i,j)
if [i,j]=[minval(B)] then
// on n'a que des zeros sur la ligne i et la colonne j sa
// en (i,j) on sauve donc B[i,j] et on raye ligne et colonn
k:=k+1; l[k]:=B[i,j]
B:=delcols(B,j..j)
B:=delrows(B,i..i)
//le cas B de taille 1 pose probleme car on ne peut pas ray
//une colonne puis une lign
fi;
// ASTUCE: voici comment assigner 2 valeurs d'un coup
(i,j):=minval(B)
end do;
// On a eventuellement change le signe du determina
diag(l[seq(l[k],k=1..n-1)],B[1,1])
end proc;
```

// Warning: k n l B minval i j trans declared as global variable(s)
// End defining Zequiv

```
proc(A)
k:=0;
n:=(dim(A))[1];
l:=seq(0,k=(1 .. n));
B:=A;
i,j:=minval(B);
while (([i,j]<>[0,0]) and ((nops(B))>1) do
B:=trans(B,i,j);
if [i,j]=[minval(B)] then
k:=k+1;
l[k]:=B[i,j];
B:=delcols(B,j .. j);
B:=delrows(B,i .. i)
fi;
i,j:=minval(B);
od;;
diag(seq(l[k],k=(1 .. (n-1))),B[1,1]);
```

end;

27 A:=matrix([[2,2,2],[6,12,6],[6,4,6]]); #Exemple:

2	2	2
6	12	6
6	4	6

```
28 Zequiv(A);
```

$$\begin{pmatrix} 2 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

```
29 A[3,3]:=12*A;
```

$$\left(\text{Done, } \begin{pmatrix} 2 & 2 & 2 \\ 6 & 12 & 6 \\ 6 & 4 & 12 \end{pmatrix} \right)$$

```
30 Zequiv(A);
```

$$\begin{pmatrix} 2 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & 18 \end{pmatrix}$$

31 On v'erifie que A et Zequiv(A) ont bien meme forme de smith:

```
32 ismith(A)[2], ismith(Zequiv(A))[2];
```

$$\left(\begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 18 \end{pmatrix}, \begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 18 \end{pmatrix} \right)$$

33 Attention, on n'obtient pas forcement les diviseurs elementaires, par exemple:

```
34 A:=matrix([[4,0,0],[0,6,0],[0,0,8]]);
```

$$\begin{pmatrix} 4 & 0 & 0 \\ 0 & 6 & 0 \\ 0 & 0 & 8 \end{pmatrix}$$

```
35 ismith(A)[2];
```

$$\begin{pmatrix} 2 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 24 \end{pmatrix}$$

```
36 Zequiv(A);
```

$$\begin{pmatrix} 4 & 0 & 0 \\ 0 & 6 & 0 \\ 0 & 0 & 8 \end{pmatrix}$$

37 ca sera forc'ement le pgcd(d_1,...,d_n) car l'algorithme reste sur la premiere ligne.

```
38 f:=(i,j)->if (i-j)*(i-1)=0 then 1 else 0 fi;
// Success
// End defining f
```

```

if (i-j)*(i-1)=0 then
1 else
0
(i, j)> fi

```

```
39 matrix(3,3,f);A:=matrix(3,3,f)*A;Zequiv(A);
```

$$\left(\begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 4 & 6 & 8 \\ 0 & 6 & 0 \\ 0 & 0 & 8 \end{pmatrix}, \begin{pmatrix} 2 & 0 & 0 \\ 0 & 8 & 0 \\ 0 & 0 & -12 \end{pmatrix} \right)$$

40 On recopie trans et Zequiv pour qu'ils ne travaillent que sur les colonnes

```

41 Prog Edit Ajouter      [nxt] [OK] [Save]
transC:=proc(A,i,j)
//attention, si on ne met pas un local n, il modifiera la valeur de n dans Zequiv
local n,U,l;
n:=dim(A)[1];
U:=identity(n);
for l from 1 to n do
U[j,l]:=-i quo(A[i,l],A[i,j])
od;
//pour ne pas mettre de test dans la boucle precedente, on n'ecarte pas l=j, mais on le corrige
U[j,j]:=1;
A*U;
end proc;

// Success
// End defining transC

proc(A,i,j)
local n,U,l;
n:=(dim(A))[1];
U:=identity(n);
for l from 1 to n+1/2 do
U[j,l]:=-i quo(A[i,l],A[i,j]);
od;
U[j,j]:=1;
A*U;
end;

```

```

42
43 Prog Edit Ajouter      [nxt] [OK] [Save]
ZequivC:=proc(A)
k:=0;
n:=dim(A)[1];
l:=seq(0,k=1..n)
B:=A;
(i,j):=minval(B);
while((i,j)<>[0,0]) and size(B)>1 do
// attention, ne pas oublier de declarer les variables locales dans trans pour eviter les melan;
B:=transC(B,i,j);
if [i,j]=[minval(B)] then
k:=k+1; l[k]:=B[i,j];
B:=delcols(B,j..j);
B:=delrows(B,i..i);
fi;
// ASTUCE: voici comment assigner 2 valeurs d'un coup
(i,j):=minval(B);
end do;
// On a eventuellement change le signe du determina;
diag([seq(l[k],k=1..n-1),B[1,1]])
end proc;

// Warning: k n | B minval i j transC declared as global variable(s)
// End defining ZequivC

proc(A)
k:=0;
n:=(dim(A))[1];
l:=seq(0,k=(1..n));
B:=A;
i,j:=minval(B);
while ((i,j)<>[0,0]) and ((nops(B))>1) do
B:=transC(B,i,j);
if [i,j]=[minval(B)] then
k:=k+1;
l[k]:=B[i,j];
B:=delcols(B,j..j);
B:=delrows(B,i..i);
fi;
i,j:=minval(B);
od;
diag(seq(l[k],k=(1..(n-1))),B[1,1]);
end;

```

45 Prog Edit Ajouter

```

elem:=proc(A)
n:=dim(A)[1];
d:=Zequiv(A);
L:=[];
for i from 1 to n-1 do
T:=matrix(n+1-i,n+1-i,f);
d:=ZequivC(T*d);
L:=[op(L),d[1,1]];
d:=delrows(delcols(d,1..1),1..1)
od;
[op(L),d[1,1]];
end proc;

```

// Warning: n Zequiv d L if T ZequivC declared as global variable(s)
// End defining elem

```

proc(A)
n:=(dim(A))[1];
d:=Zequiv(A);
L:=[];
for i from 1 to n-1+1/2 do
T:=matrix(n+1-i,n+1-i,f);
d:=ZequivC(T*d);
L:=[op(L),d[1,1]];
d:=delrows(delcols(d,1..1),1..1);
od;
[op(L),d[1,1]];
end;

```

46 A:=matrix([[2,2,2],[6,12,6],[6,4,12]]);

$$\begin{pmatrix} 2 & 2 & 2 \\ 6 & 12 & 6 \\ 6 & 4 & 12 \end{pmatrix}$$

47 elem(A);ismith(A)[2];

$$\left([2 \ -2 \ 18 \], \begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 18 \end{pmatrix} \right)$$

48 elem(diag(4,6,16));ismith(diag(4,6,16))[2];

$$\left([2 \ 4 \ 48 \], \begin{pmatrix} 2 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 48 \end{pmatrix} \right)$$