1 restart;maple_mode(1);cas_setup(0,0,0,1,0,1e-10,25,[1,50,0,25],0,0,0);#radians,pas de cmplx, pas de Sqrt

Warning: some commands like subs might change arguments order

Menu

2 diag(seq(1,4)); diag(seq(i,i=1..4));

$$\left(\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix},\begin{bmatrix}1&0&0&0\\0&2&0&0\\0&0&3&0\\0&0&0&4\end{bmatrix}\right)$$

Menu

3 A:=matrix(4,4)+1;v:=[seq(1,j=1..4)];

$$\left(\begin{bmatrix}1&1&1&1\\1&1&1&1\\1&1&1&1\\1&1&1&1\end{bmatrix},\begin{bmatrix}1&1&1&1\end{bmatrix}\right)$$

Menu

4 A*v;# Attention il retourne une ligne

$$\begin{bmatrix}4&4&4&4\end{bmatrix}$$

Menu

5 purge(a);

No such variable a

Menu

6 f:=(i,j)->if (i=j) then 0 else if (i<j) then a[i,j] else -a[j,i] fi;fi;

// Warning: a declared as global variable(s)
// End defining f

```
          if i=j then
           0 else
           if i<j then
            a[i,j] else
            -(a[j,i])
           fi
( i,  j  )->  fi
```

Menu

7 matrix(4,4,f); d:=det(matrix(4,4,f)):

$$\left(\begin{bmatrix}0&a\llbracket 1,2\rrbracket&a\llbracket 1,3\rrbracket&a\llbracket 1,4\rrbracket\\-(a\llbracket 1,2\rrbracket)&0&a\llbracket 2,3\rrbracket&a\llbracket 2,4\rrbracket\\-(a\llbracket 1,3\rrbracket)&-(a\llbracket 2,3\rrbracket)&0&a\llbracket 3,4\rrbracket\\-(a\llbracket 1,4\rrbracket)&-(a\llbracket 2,4\rrbracket)&-(a\llbracket 3,4\rrbracket)&0\end{bmatrix},\ \text{Done}\right)$$

Menu

8 factor(d);#c'est toujours un carre

$$\left((a\llbracket 1,2\rrbracket)\cdot(a\llbracket 3,4\rrbracket)-(a\llbracket 1,3\rrbracket)\cdot(a\llbracket 2,4\rrbracket)+(a\llbracket 1,4\rrbracket)\cdot(a\llbracket 2,3\rrbracket)\right)^2$$

Menu

9 M:=matrix(8,8,f);

$$\begin{bmatrix}0&a\llbracket 1,2\rrbracket&a\llbracket 1,3\rrbracket&a\llbracket 1,4\rrbracket&a\llbracket 1,5\rrbracket&a\llbracket 1,6\rrbracket&a\llbracket 1,7\rrbracket\\-(a\llbracket 1,2\rrbracket)&0&a\llbracket 2,3\rrbracket&a\llbracket 2,4\rrbracket&a\llbracket 2,5\rrbracket&a\llbracket 2,6\rrbracket&a\llbracket 2,7\rrbracket\\-(a\llbracket 1,3\rrbracket)&-(a\llbracket 2,3\rrbracket)&0&a\llbracket 3,4\rrbracket&a\llbracket 3,5\rrbracket&a\llbracket 3,6\rrbracket&a\llbracket 3,7\rrbracket\\-(a\llbracket 1,4\rrbracket)&-(a\llbracket 2,4\rrbracket)&-(a\llbracket 3,4\rrbracket)&0&a\llbracket 4,5\rrbracket&a\llbracket 4,6\rrbracket&a\llbracket 4,7\rrbracket\\-(a\llbracket 1,5\rrbracket)&-(a\llbracket 2,5\rrbracket)&-(a\llbracket 3,5\rrbracket)&-(a\llbracket 4,5\rrbracket)&0&a\llbracket 5,6\rrbracket&a\llbracket 5,7\rrbracket\\-(a\llbracket 1,6\rrbracket)&-(a\llbracket 2,6\rrbracket)&-(a\llbracket 3,6\rrbracket)&-(a\llbracket 4,6\rrbracket)&-(a\llbracket 5,6\rrbracket)&0&a\llbracket 6,7\rrbracket\\-(a\llbracket 1,7\rrbracket)&-(a\llbracket 2,7\rrbracket)&-(a\llbracket 3,7\rrbracket)&-(a\llbracket 4,7\rrbracket)&-(a\llbracket 5,7\rrbracket)&-(a\llbracket 6,7\rrbracket)&0\\-(a\llbracket 1,8\rrbracket)&-(a\llbracket 2,8\rrbracket)&-(a\llbracket 3,8\rrbracket)&-(a\llbracket 4,8\rrbracket)&-(a\llbracket 5,8\rrbracket)&-(a\llbracket 6,8\rrbracket)&-(a\llbracket 7,8\rrbracket)\end{bmatrix}$$

Menu

10 Methode type pivot, contre m{'e}thode de Laplace,... mais ici la matrice
est antisymetrique.

11 d1:=det(M):

Evaluation time: 5.16

Done

Menu

12 d2:=det_minor(M):

Done

13 normal(d1-d2);

Evaluation time: 2.31

0

Menu

14 ------------------Illustration de la reduction de Jordan--

Construction de l'exemple: on veut une reponse de ce type:

15 f:=(i,j)->if(i=j-1) then 1 else 0 fi;

// Success
// End defining f

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

Menu

16 J:=matrix(8,8,f);#forme classique d'ordre 8.

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

Menu

17 J[3,4]:=0:J[6,7]:=0:J;#2 blocs d'ordre 3 et un d'ordre 2.

$$\left(\text{Done}, \text{Done}, \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}\right)$$

Menu

18 On veut faire un changement de base simple. Ex det=1 pour garder des coeffs entiers.
on cree une transvection:

19 f:=(i,j)->if(i=j) then 1 else 0 fi;

// Success
// End defining f

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

Menu

20 T:=proc(i,j,a)
local A;
A:=matrix(8,8,f):A[i,j]:=a;A;
end proc;

// Warning: f declared as global variable(s)
// End defining T

```
proc(i,j,a)
  local A;
  A:=matrix(8,8,f);
  A[i,j]:=a;
  A;
```

21 B:=matrix(8,8,b);

$$\begin{bmatrix} b & b & b & b & b & b & b & b \\ b & b & b & b & b & b & b & b \\ b & b & b & b & b & b & b & b \\ b & b & b & b & b & b & b & b \\ b & b & b & b & b & b & b & b \\ b & b & b & b & b & b & b & b \\ b & b & b & b & b & b & b & b \\ b & b & b & b & b & b & b & b \end{bmatrix}$$

Menu

22 faire Li<-Li+aLj c'est multiplier a gauche par T(i,j,a).
Par exemple L3<- L3+aL2 c'est multiplier a GAUCHE par: T(3,2,a);

23 T(3,2,a)*B;

$$\begin{bmatrix} b & b & b & b & b & b & b & b \\ b & b & b & b & b & b & b & b \\ a\cdot b+b & a\cdot b+b & a\cdot b+b & a\cdot b+b & a\cdot b+b & a\cdot b+b & a\cdot b+b & a\cdot b+b \\ b & b & b & b & b & b & b & b \\ b & b & b & b & b & b & b & b \\ b & b & b & b & b & b & b & b \\ b & b & b & b & b & b & b & b \\ b & b & b & b & b & b & b & b \end{bmatrix}$$

Menu

24 En revanche: C2<-C2+aC3 c'est multplier a DROITE par T(3,2,a)

25 B*T(3,2,a);

$$\begin{bmatrix} b & a\cdot b+b & b & b & b & b & b & b \\ b & a\cdot b+b & b & b & b & b & b & b \\ b & a\cdot b+b & b & b & b & b & b & b \\ b & a\cdot b+b & b & b & b & b & b & b \\ b & a\cdot b+b & b & b & b & b & b & b \\ b & a\cdot b+b & b & b & b & b & b & b \\ b & a\cdot b+b & b & b & b & b & b & b \\ b & a\cdot b+b & b & b & b & b & b & b \end{bmatrix}$$

Menu

26 Remarquer que l'inverse de T(i,j,a) est T(i,j,-a)

27 T(3,2,a)^(-1);

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -a & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Menu

28 Donc conjuguer par T(i,j,a) c'est faire:
Li <- Li+aLj et Cj <- Cj-aCi

29 P:=T(6,7,2)*T(4,5,1)*T(3,2,2)*T(1,2,1):

Done

30 P,P^(-1);

$$\left(\begin{bmatrix}1&1&0&0&0&0&0&0\\0&1&0&0&0&0&0&0\\0&2&1&0&0&0&0&0\\0&0&0&1&1&0&0&0\\0&0&0&0&1&0&0&0\\0&0&0&0&0&1&2&0\\0&0&0&0&0&0&1&0\\0&0&0&0&0&0&0&1\end{bmatrix}, \begin{bmatrix}1&-1&0&0&0&0&0&0\\0&1&0&0&0&0&0&0\\0&-2&1&0&0&0&0&0\\0&0&0&1&-1&0&0&0\\0&0&0&0&1&0&0&0\\0&0&0&0&0&1&-2&0\\0&0&0&0&0&0&1&0\\0&0&0&0&0&0&0&1\end{bmatrix}\right)$$

Menu

31 Donc faire a l'ordinateur:

32 N:=P*J*P^(-1):

Done

Menu

33 est identique a faire a la main a partir de J:
L1 <- L1+L2 ; C2<-C2 - C1 puis
L3 <- L3+2L2 ; C2 <- C2 -2C3
L4 <- L4+L5 ; C5 <- C5 -C4
L6 <- L6 +2 L7; C7 <- C7 -2 C6
On a maintenant trouve un bel exercice: Trouver la forme de jordan de N et une matrice de passage pour l'obtenir.
On calcule N^2 et son noyau.

34 N,N^2;

$$\left(\begin{bmatrix}0&-1&1&0&0&0&0&0\\0&-2&1&0&0&0&0&0\\0&-4&2&0&0&0&0&0\\0&0&0&0&1&1&-2&0\\0&0&0&0&0&1&-2&0\\0&0&0&0&0&0&0&2\\0&0&0&0&0&0&0&1\\0&0&0&0&0&0&0&0\end{bmatrix}, \begin{bmatrix}0&-2&1&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&1&-2&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\end{bmatrix}\right)$$

Menu

35 N2:=nullspace(N^2);

$$\begin{bmatrix}-1&0&0&0&0&0&0&0\\0&\frac{-1}{2}&-1&0&0&0&0&0\\0&0&0&-1&0&0&0&0\\0&0&0&0&-1&0&0&0\\0&0&0&0&0&-2&-1&0\\0&0&0&0&0&0&0&-1\end{bmatrix}$$

Menu

36 on choisit a et b independants modulo ker N^2 (qui est aussi im N). (attention a et b hors de ker N^2 est insuffisant).

37 a:=[0,0,1,0,0,0,0,0]:b:=[0,0,0,0,0,1,0,0]:

(Done, Done)

Menu

38 rank(matrix([op(N2),a,b]));# doit etre dim ker N^2 +2.

8

Menu

39 N1:=nullspace(N);

$$\begin{bmatrix}-1&0&0&0&0&0&0&0\\0&0&0&-1&0&0&0&0\end{bmatrix}$$

40 dim ker N^2 -dim ker N= 6-3=2+1 donc N.a,N.b doit etre complete par c
tq N.a,N.b,c indep modulo ker N. Par exemple on prend celui la:

41 c:=[0,0,0,0,0,0,0,1]:

Done

Menu

42 On verifie qu'il convient:

43 rank(matrix([op(N1),N*a,N*b,c]));

6

Menu

44 dim ker N - dim ker N^0=3 c'est donc engendr{'e} par
N^2.a,N^2.b,N.c. Il n'y a plus rien a faire, et l'on prend
la base suivante: (Attention pour xcas N*a... sont des lignes, on
prend donc la transposee)

45 Q:=transpose(matrix([(N^2)*a,N*a,a,(N^2)*b,N*b,b,N*c,c]));

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Menu

46 On sait maintenant que Q^(-1)*N*Q doit donner J. verifification:

47 Q^(-1)*N*Q;

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

Menu

48 --

49 Il s'agit donc de trouver les bases qui jordanisent l'endomorphisme.
On a N^3=0.
Il nous faut d'abord pour les 2 blocs de taille 3, une base de kerN^3/ker N^2
que l'on remonte ensuite dans ker N^3 (Choix de a et b dans la question
precedente). Comme ker N^3/ker N^2 est de dimension
2, on a deja: card(GL2).(card ker N^2)^2. qui vaut: (p^2-1)*(p^2-p)*(p^6)^2.
reste le choix d'un vecteur non nul de ker N^2/(ker N +Im N) que l'on remonte.
ker N+im N est de dim 3+2. On a donc p-1 choix dans le quotient, soit (p-1)*p^5
choix pour le dernier vecteur a choisir (c dans la question precedente).

50 cardstab:=(p^2-1)*(p^2-p)*p^12*(p-1)*p^5

$$(p^2-1)\cdot(p^2-p)\cdot p^{12}\cdot(p-1)\cdot p^5$$

Menu

51 Autre methode: On commence par choisir les vecteurs de base qui sont dans le
noyau. On prend une base de Im N^2 qui est de dim 2, soit (p^2-1)*(p^2-p) puis
un vecteur du noyau hors de Im N^2. soit (p^3-p^2) choix. On cherche alors des
antecedants a ces vecteurs. (p^3 choix pour l'antecedant d'un vecteur)

52 cardstab2:=(p^2-1)*(p^2-p)*(p^3-p^2)*(p^3)^3*(p^3)^2;

$$(p^2-1)\cdot(p^2-p)\cdot(p^3-p^2)\cdot p^9\cdot p^6$$

53 normal(cardstab-cardstab2);

0

54 cardGL8:=product((p^8-p^i),i=0..7);

$(p^8-1)\cdot(p^8-p)\cdot(p^8-p^2)\cdot(p^8-p^3)\cdot(p^8-p^4)\cdot(p^8-p^5)\cdot(p^8-p^6)\cdot(p^8-p^7)$

55 cardorbite:=normal(cardGL8/cardstab);

$p^{42}+p^{41}+p^{40}-p^{38}+(-2)\cdot p^{37}+(-3)\cdot p^{36}+(-3)\cdot p^{35}+(-3)\cdot p^{34}-p^{33}+p^{32}+4\cdot p^{31}+5\cdot p^{30}+6\cdot p^{29}+5\cdot p^{28}$
$(-4)\cdot p^{21}-p^{20}+p^{19}+3\cdot p^{18}+3\cdot p^{17}+3\cdot p^{16}+2\cdot p^{15}+p^{14}-p^{12}-p^{11}-p^{10}$

56

57 ---Exercice---syntaxe et combinatoire des matrice nilpotentes--------------------------------

58 n:=5;purge(a);

(5, [0 0 1 0 0 0 0 0])

59 A:=matrix(n,n,(i,j)->a[n*(i-1)+j]);

// Warning: a n declared as global variable(s)

$$\begin{bmatrix} a[1] & a[2] & a[3] & a[4] & a[5] \\ a[6] & a[7] & a[8] & a[9] & a[10] \\ a[11] & a[12] & a[13] & a[14] & a[15] \\ a[16] & a[17] & a[18] & a[19] & a[20] \\ a[21] & a[22] & a[23] & a[24] & a[25] \end{bmatrix}$$

60 l:=[seq(a[i],i=1..n^2)];

[a[1] a[2] a[3] a[4] a[5] a[6] a[7] a[8] a[9] a[10] a[11] a[12] a[13] a[14] a[15] a[16] a

61 Pour convertir la liste l en une matrice a n colonne:

62 A:=list2mat(l,n);

$$\begin{bmatrix} a[1] & a[2] & a[3] & a[4] & a[5] \\ a[6] & a[7] & a[8] & a[9] & a[10] \\ a[11] & a[12] & a[13] & a[14] & a[15] \\ a[16] & a[17] & a[18] & a[19] & a[20] \\ a[21] & a[22] & a[23] & a[24] & a[25] \end{bmatrix}$$

63 J:=matrix(n,n):J[1,2]:=1:J[3,4]:=1:J;

$$\left(\text{Done}, \text{Done}, \text{Done}, \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \right)$$

64 L:=mat2list(A*J-J*A);

[- (a[6]) a[1] - (a[7]) - (a[8]) a[3] - (a[9]) - (a[10]) 0 a[6] 0 a[8] 0 - (a[16]) a[11] - (a[17]) - (a

65 C:=matrix(n^2,n^2,(i,j)->diff(L[i],a[j]));

// Warning: L a declared as global variable(s)

$$\begin{bmatrix}
0&0&0&0&0&-1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&0&0&0&0&-1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&-1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0&-1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&-1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&-1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&-1&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&-1&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&-1&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&-1&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0
\end{bmatrix}$$

Menu

66 n^2-rank(C); #est la dim du commutateur de J.

13

Menu

67 Nullspace(matrix([[2]])) mod 2; #est correct

$[-1]$

Menu

68 nullspace(matrix([[2]])) mod 2; #cherche d'abord le noyau sur Q puis le reduit mod2

[]

Menu

69 maple_mode(0); # en mode xcas il suffit d'utiliser %, on reste dans Z/2Z

Warning: some commands like subs might change arguments order

Menu

70 e2:=1 % 2; maple_mode(1);

(1 % 2, Warning: some commands like subs might change arguments order)

Menu

71 nullspace(matrix([[2*e2]]));

$[-1]$

72 `com:=Nullspace(C) mod 2;`

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

Menu

73 `COM:=[seq(list2mat(i,n),i=com)];`

$$\left[\begin{bmatrix}0&1&0&0&0\\0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\end{bmatrix}\begin{bmatrix}0&0&0&1&0\\0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\end{bmatrix}\begin{bmatrix}0&0&0&0&1\\0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\end{bmatrix}\begin{bmatrix}1&0&0&0&0\\0&1&0&0&0\\0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\end{bmatrix}\begin{bmatrix}0&0&1&0&0\\0&0&0&1&0\\0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\end{bmatrix}\begin{bmatrix}0&0\\0&0\\0&1\\0&0\\0&0\end{matrix}\right.$$

Menu

74 `v:=[1$13];`

$$[1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1]$$

Menu

75 `v*COM;`

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

Menu

76 `[1,2,3]*[a,b,c,d,e];`

$$[[2\quad 0\quad a+3\ ]]$$

Menu

77 `convert(67,base,2);`

$$[1\ 1\ 0\ 0\ 0\ 0\ 1]$$

Menu

78 `convert(0,base,2);# Attention, il ne retoure pas 0 mais le vide.`

[]

Menu

79 `augment(convert(67,base,2),[0$10]); # pour ajouter des 0`

$$[1\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$$

Menu

80 `LCOM:=[];`

[]

Menu

81
```
for i from 0 to 2^13-1 do
ci:=convert(i,base,2);LCOM:=[op(LCOM),augment(ci,[0$(13-dim(ci))])*COM] od;
```

Evaluation time: 55.81

Done

82 `STAB:=select(x->(det(x) mod 2)<>0,LCOM);`

// Success

83 dim(STAB);

1536

Menu

84 Le cardinal du stabilisateur est aussi le nombre de bases qui
Jordanisent l'endomorphisme J. de partition:
XX qui represente e1e2
XX e3e4
X e5
on choisit d'abord les classes de e2 et e4 dans kerJ^2/kerJ qui est de dim 2, ce qui fait 3*2 choix (le cardinal de GL2(F2)). puis on remonte ces classes dans kerJ^2. (Chaque classe a autant d'antecedants que kerJ:2^3. On a donc 6*8*8 choix pour e2,e4. ils determinent e1 et e3. On choisit e5 dans ker N-(vect(e1,e3)):8-4 choix. Au total: 6*8*8*4=1536

85