

Algorithme de LE VERRIER

(calcul du polynôme caractéristique d'une matrice)

Pour $A \in \mathcal{M}_n(\mathbb{K})$, on pose : $A_0 = A$ et $\forall k \in \{1, \dots, n\}$ $A_k = A \times \left[A_{k-1} - \frac{1}{k} \text{Tr}(A_{k-1}) \cdot I_n \right]$.

Alors : $\chi_A(X) = X^n - \sum_{k=1}^n \frac{1}{k} \text{Tr}(A_{k-1}) \cdot X^{n-k}$.

Avec les calculatrices ou les logiciels qui ne font pas de calcul formel, on peut grâce à cet algorithme calculer **les coefficients** du polynôme caractéristique.

D'où le programme pour TI 8x ($x < 5 \dots$), qui remplit la liste L1 avec les coefficients de χ_A et l'affiche :

```
PROGRAM:PCAR
:Ans→[A]
:Ans→[J]
:dim([A])→L6
:L6(1)→N
:identity(N)→[I]
:N+1→dim(L1)
:1→L1(1)
:For(K,1,N)
:-sum(seq([J](I,I),I,1,N))/K→L1(K+1)
:[A]*([J]+Ans*[I])→[J]
:End
:L1
```

La Casio 35+ n'accepte pas toutes les commandes sur les matrices dans les programmes.

Le programme suivant calcule et affiche l'un après l'autre les coefficients de χ_A , à condition que A (*resp.* I_n) soit au préalable stockée dans Mat A (*resp.* Mat I).

```
"TAILLE DE A " ?→N
Mat A→Mat J
1▲
For 1→K To N
0→T
For 1→I To N:T+Mat J[I,I]→T:Next
-T÷K▲
Mat A×(Mat J-T÷K×Mat I)→Mat J
Next
Return
```

Avec Python, le module `numpy` permet de mettre en œuvre facilement cet algorithme, mais on obtient des valeurs numériques approchées des coefficients (ce qui n'est pas gênant lorsque ce sont des entiers...).

```
import numpy as np

def khi(A):
    n=len(A)
    I=np.eye(n)
    Ak=A
    P=[1]
    for k in range(1,n+1):
        c=-np.trace(Ak)/k
        P.append(c)
        Ak=np.dot(A,Ak+c*I)
    return P
```

`len(A)` renvoie le nombre de lignes de A.

`numpy` fournit aussi la fonction `shape` qui renvoie le **couple des dimensions** du tableau, ainsi `np.shape(A)[0]` donnera également le nombre de lignes de A.

La fonction `trace` se passe de commentaire, `eye` renvoie l'identité.

Le produit matriciel de deux tableaux `numpy` se calcule grâce à la fonction `dot`.

Noter qu'il existe déjà dans `numpy` une fonction qui calcule les coefficients du polynôme caractéristique d'une matrice carrée : `np.poly(A)` renvoie un vecteur contenant lesdits coefficients, ordonnés par degré décroissant.

Noter enfin que le module `sympy` permet de faire du calcul formel avec Python, de même qu'avec certaines calculatrices et certains logiciels comme Maple, Mathematica... Mais tout cela est hors programme !