

Erreurs et remarques

Ce sujet 0 est a été publié sur le site du concours CCP. Il comporte quelques erreurs et manques, sans conséquences sur la compréhension et la notation de l'épreuve.

Page 3/11 : erreurs dans les équations, sans conséquences pour la suite du sujet

$$m_i \frac{d^2 u_i(t)}{dt^2} = -k_i (u_i(t) - u_{i-1}(t)) - k_{i+1} (u_i(t) - u_{i+1}(t)) - c_i (\dot{u}_i(t) - \dot{u}_{i-1}(t)) - c_{i+1} (\dot{u}_i(t) - \dot{u}_{i+1}(t))$$

$$m_1 \frac{d^2 u_1(t)}{dt^2} = -(k_1 + k_2) u_1(t) + k_2 u_2(t) - (c_1 + c_2) \dot{u}_1(t) + c_2 \dot{u}_2(t)$$

$$m_n \frac{d^2 u_n(t)}{dt^2} = -k_n (u_n(t) - u_{n-1}(t)) - c_n (\dot{u}_n(t) - \dot{u}_{n-1}(t)) + f_n(t)$$

Page 4/11 : programme en Python. Il manque quelques arguments aux fonctions, sans conséquences pour la suite du sujet.

■ Python

```
X=calcul(n,M,K,C,npts,dt,fmax,omega)
posttraitement(X,L,n,c,dt)
```

Page 9/11 : erreur dans la formule de la puissance dissipée :

$$P_{\text{diss}} = C_1 (\dot{u}_1(t))^2 + \sum_{i=2}^n c_i (\dot{u}_i(t) - \dot{u}_{i-1}(t))^2$$

Q15, un argument supplémentaire est nécessaire : `Ediss=calcul_energie(X,c,dt)`.

Page 10/11 : méthode `readline()`

Une ligne est une chaîne de caractères qui se termine par le caractère ASCII « retour ligne » (Line Feed : '\n'). La méthode `readline()` lit et retourne une ligne. `float('12\n')` retourne le réel 12.0.

Page 11/11 : Python et numpy : on suppose que pour tout le sujet, la bibliothèque numpy a été importée.

- La définition d'un tableau peut se faire de plusieurs manières :
 - Python : `T=[[1,2,3],[2,1,5],[4,5,8]]` ;
 - Numpy : `T=array([[1,2,3],[2,1,5],[4,5,8]])`.

On peut alors accéder à des fonctionnalités supplémentaires. Par exemple : `T[i,j]` identique à `T[i][j]`. `T[:,0:3]` : extrait les trois premières colonnes. `T=zeros((3,3),float)` : T est de type 'array'.

- Le produit matriciel n'est pas « * » pour les tableaux. Il existe pour cela la fonction numpy « dot ». Par exemple : `B = dot(A,X)` (`B=AX` où A est une matrice, X un vecteur)
 - Il est possible avec numpy de créer une variable de type 'matrix' (matrice) qui admet alors l'opérateur « * » pour produit, à la manière de Scilab ou Matlab. Il existe alors aussi un certain nombre d'opérateurs et de méthodes associés au calcul matriciel. Par exemple : `M=matrix([1,2,3],[2,1,5],[4,5,8])` M.T : transposée de la matrice M `M**2 : M*M`.
 - Pour effectuer une copie par valeurs et non par référence d'un tableau, il existe la fonction numpy « copy ». Par exemple : `A = copy(B)`. La modification de A, n'aura alors aucune conséquence sur B. Cela n'est pas le cas si l'instruction « `A = B` » a été tapée.
- Bibliothèque « matplotlib.pyplot » : On suppose que pour tout le sujet, la bibliothèque matplotlib.pyplot a été importée.