ECE 1 - Année 2015-2016
Lycée français de Vienne
Mathématiques - F. Gaunard
http://frederic.gaunard.com

Informatique - T.P n°3

On commencera par créer, dans son dossier personnel, un dossier intitulé "TP3" dans lequel on pourra enregistrer tous les programmes (au format *.sce*) ou fonctions (au format *.sci*) relatifs à cette nouvelle feuille d'exercices.

Il peut être intéressant de représenter graphiquement, avec `SciLab`*, une suite réelle* (u_n)*. Pour cela, on utilise la fonction* `plot2d()`*. Plus précisément,* `plot2d(X,Y,-1)` *trace une croix aux points de coordonées* (x_k, y_k) *où* `X=[`$x_1, ..., x_n$`]` *et* `Y=[`$y_1, ..., y_n$`]` *sont des vecteurs-ligne de même longueur. En remplaçant -1 par -2 ou -3, on change le style et la taille de la croix.*

Exercice 1. (Représentation graphique d'une suite)

(1) Écrire une suite d'instruction pour obtenir un vecteur-ligne dont les 25 composantes correspondent au 25 premiers termes de la suites (u_n) définie pour $n \geq 1$ par

$$u_n = \frac{\ln(\sqrt{n}+1)}{\ln((n+1)^2-2)}.$$

(2) Représenter graphiquement, avec l'aide de la fonction `plot2d()` les 25 premiers termes de la suite (u_n). Que peut-on conjecturer? Le démontrer.

En tapant la commande `plot2d(X,Y)`, `SciLab` *trace simplement la ligne brisée formée des points de coordonnées* (x_k, y_k)*. Si* `Z` *est un vecteur de même longueur que* `X` *et* `Y`*, on peut représenter la ligne brisée formée des points de coordonnées* (x_k, z_k) *sur un même graphique que la ligne brisée précédente avec l'instruction* `plot2d(X, [Y,Z])` *(ceci marche également pour davantage de lignes brisées). Cependant, cette commande ne marche qu'avec des vecteurs* ***colonne****! Bonne nouvelle: la commande* `Y'` *transforme le vecteur-ligne* `Y` *en un vecteur colonne.*

La commande `linspace(a,b,n)` *crée un vecteur ligne de longueur* n *dont les composantes découpent l'intervalle* $[a; b]$ *en* $n-1$ *segments (subdivisions) de même longueur.*

Exercice 2. (Itérations) On considère la fonction f définie sur $\mathbb{R}+$ par $f(x) = \frac{1}{1+x}$.

(1) Écrire une fonction `f-exo2()` prenant pour argument un réel positif x et renvoyant la valeur de $f(x)$.

(2) Définir `X=linspace(0,1,11)` puis un vecteur colonne `Y1` tel que `Y(i)=f-exo2(X(i))`. À l'aide de la fonction `plot2d()`, représenter la fonction f sur l'intervalle $[0; 1]$.

(3) Écrire une fonction `iterf()` prenant pour argument**s** un réel positif x et un entier $n \geq 1$ et renvoyant la $n-$ième itérée de f évaluée en x : $f \circ f \circ \cdots \circ f(x)$.

(4) Définit des vecteurs colonnes `Y2`, `Y3`, `Y5` et `Y10` analogues à `Y1` mais correspondant aux itérées de f pour les valeurs respectives de n égales à 2, 3, 5 et 10. Utiliser la fonction `plot2d()` pour les représenter toutes sur un même graphique. Que semble-t-on constater?

Exercice 3. (Combinatoire)

(1) Écrire une fonction `facto()` prenant en argument un entier n et renvoyant $n!$. (On pourra notamment utiliser la fonction `prod()` qui s'utilise comme la fonction `sum()`.)

(2) Utiliser cette fonction pour en écrire deux autres, nommées `arrangement()` et `combi()`, prenant toutes deux pour arguments n et k, où n est un entier naturel et k un entier compris entre 0 et n.

(3) Vérifier avec `SciLab` que

$$\sum_{k=0}^{n} \binom{n}{k} = 2^n \quad \text{et} \quad \sum_{k=0}^{n} (-1)^k \binom{n}{k} = 0.$$

Exercice 4. (Triangle de Pascal)

(1) Rappeler la formule du triangle de Pascal.

(2) Écrire une fonction `triangle()` prenant en argument un vecteur ligne **c** (de longueur n) et renvoyant un vecteur ligne `d` de longueur $n+1$ tel que

`d(1)=d(n+1)=c(1)` et `d(k)=c(k-1)+c(k)`.

(3) Écrire alors un programme qui affiche les m premières lignes du triangles de Pascal, où m est un entier positif rentré par l'utilisateur.