La fonction Nest

La suite (a_n) définie dans le TP2 est définie par itération de la fonction $f(t)=\frac{1}{2}(t+\frac{2}{t})$. (i.e. $a_{n+1}=f(a_n),\ a_0=1$).

Pour calculer la valeur de $\overbrace{f\circ f\circ\cdots\circ f}^{n\text{ fois}}(a)$ pour une fonction f à partir de la valeur initiale a, *Mathematica* peut réaliser ces calculs à l'aide de la fonction `Nest`.
Pour conserver l'ensemble des valeurs $(f(a), f\circ f(a), \dots, f^{(n)}(a))$ dans une liste on utilise alors la fonction `NestList`. Évaluer les entrées suivantes :

```
Nest[f,x0,3]
NestList[f,x0,3]
```

1. En définissant `f[x_]:=1/2*(x+2/x)`, programmer le calcul des premiers termes de la suite (a_n) à l'aide de la fonction `Nest`.
   Comparer `NestList[f,1,4]` et `NestList[f,1.,4]`
2. À l'aide de la fonction `Nest`, programmer une fonction permettant de calculer les termes des suites (x_n) et (y_n) définies par les relations :
$$x_0=a,\ y_0=b,\ x_{n+1}=\frac{x_n+y_n}{2},\ y_{n+1}=\sqrt{x_ny_n}.$$
   (indication : on utilisera la fonction $(x,y)\mapsto(\frac{x+y}{2},\sqrt{xy})$. On écrira `f[{x_,y_}]:=...`*)*
3. *(Extrait oraux Centrale 2007, voir TP2)*
   Soit $(x_n)_{n\geq 1}$ définie par $x_1=x$ et $\forall n\in\mathbb{N}^*$, $x_{n+1}=x_n+n/x_n$.
   Reprogrammer le calcul des termes de cette suite à l'aide de la fonction `Nest`.
   (on pourra utiliser la fonction $(x,n)\mapsto\dots\dots$)
4. Utiliser la fonction `Nest` pour calculer les premiers termes de la suite de Fibonacci. (i.e. $u_1=1,\ u_2=1,$ et $\forall n\in\mathbb{N}^*,\ u_{n+2}=u_{n+1}+u_n$)

Exercices divers, au choix

1. Pour tout x réel, on appelle *arrondi* de x le réel $\alpha(x)=\lfloor x+\frac{1}{2}\rfloor$. On pose $\delta(x)=|x-\alpha(x)|$.
   (a) Tracer le graphe de la fonction α, de la fonction δ.
   (b) Tracer la graphe, pour différentes valeurs de l'entier n, de la fonction
$$x\mapsto\sum_{k=0}^{n}\frac{\delta(2^kx)}{2^k}.$$
2. La fonction `RandomInteger` permet de simuler le « tirage au sort » d'un entier (par exemple `RandomInteger[15]` simule le tirage au sort d'un entier de $\{0,1,\dots,15\}$).
   À l'aide de `RandomInteger`, construire une fonction `hasard[p_,n_]` permettant de tirer au hasard p entiers distincts de $\{1,2,\dots,n\}$.
   (on pourra sélectionner au hasard un entier de la liste $\{1,\dots,n\}$, extraire cet entier de la liste, sélectionner un 2e entier de cette nouvelle liste, ainsi de suite .... Voir la fct `Delete`*.)*
3. (a) Vérifier que pour plusieurs valeurs d'entiers naturels n supérieur ou égal à 2, le nombre n^5+n^4+1 n'est pas premier.
   (b) **Prouver** que quel que soit n supérieur ou égal à 2, n^5+n^4+1 n'est pas premier.
4. *(Extrait Centrale 2011)*
   À l'aide de *Mathematica* déterminer le plus petit entier naturel non nul k tel que $10^k\equiv 1[49]$ puis, à l'aide de *Mathematica*, évaluer
$$\sum_{k=1}^{42}\cos(2\pi 10^k/49).$$
   Conjecturer le résultat. Le démontrer (« à la main »).
5. *(Extrait Centrale 2011)*
   On se place dans le plan euclidien canonique.

(a) Soient H, I, J trois points distincts. Programmer un fonction permettant de calculer les coordonnées du projeté orthogonal de H sur (IJ). Les points étant donnés par leur coordonnées dans un repère orthonormé, on écrira :

`f[{h1_,h2_},{i1_,i2_},{j1_,j2_}]:=......`

(b) Soient A, B, C trois points non alignés. Si $M \in (AB)$, on note P_1 le projeté orthogonal de M sur (BC), P_2 le projeté orthogonal de P_1 sur (AC), et N le projeté orthogonal de P_2 sur (AB). Programmer une fonction ayant A, B, C, M pour paramètres d'entrées et qui retourne N.

(Les point A, B, C, M seront à nouveau donnés par leurs coordonnées)

(c) On munit la droite (AB) d'un repère. Si M a pour coordonnée x, on note $\varphi(x)$ la coordonnée de N. On considère alors la suite (x_n) définie par son premier terme x_0 et la relation $x_{n+1} = \varphi(x_n)$. Vérifier expérimentalement sur plusieurs exemples que la suite (x_n) converge.