ÉPREUVE D'INFORMATIQUE (PYTHON) №2
DURÉE : 2H

Dr. A. Ammar - IPEST : HAP 29 janvier 2021

Documents non autorisés.

— L'épreuve comporte 3 pages.

— La présentation, la lisibilité, la qualité de la rédaction et la précision des raisonnements entreront pour une part importante dans l'appréciation des copies.

— Les algorithmes doivent être commentés.

Exercice 1 : Générer des coordonnées équidistantes

Nous voulons générer $n+1$ coordonnées x équidistantes dans $[a,b]$.

Stocker, pour `a = -2`; `b = 3` et `n= 20` les coordonnées x dans une liste `xList`.

Q1. Définir toutes les variables puis utiliser une boucle **for** et ajouter chaque coordonnée à la liste `xList` (*initialement vide*).

Indications.
Avec n intervalles, correspondant à $n+1$ points, dans $[a,b]$, chaque intervalle a une longueur $h=(b-a)/n$. Les coordonnées peuvent alors être générées par la formule `xi = a + i * h`; $i=0,,n$.

Q2. Utiliser une liste de compréhension comme une implémentation alternative.

Q3. Vectoriser la liste résultante `xList` en un tableau numpy `xVect`.
N'oubliez pas **d'importer** d'abord la fonction qui transforme les listes en tableaux à partir de `numpy`.

Exercice 2 : Courbes paramétriques

Écrire les programmes Python qui tracent les courbes $y=f(x)$ suivantes :

Q1. La Lemniscate de Bernoulli $\begin{cases} x(t) = \dfrac{sin(t)}{1+cos^2(t)} \\ y(t) = \dfrac{sin(t)\times cos(t)}{1+cos^2(t)} \end{cases}$ pour $t\in[0,2\pi]$.

Q2. La spirale d'Archimède $\begin{cases} x(t) = t\times cos(t) \\ y(t) = t\times sin(t) \end{cases}$ pour $t\in[0,10\pi]$.

Q3. La courbe du cœur $\begin{cases} x(t) = 6 \times sin^3(t) \\ y(t) = cos(t) - 5 \times cos(2t) - 2 \times cos(3t) - cos(4t) \end{cases}$ pour $t \in [0, 2\pi]$.

Q4. Les cyclo-harmoniques $\begin{cases} x(t) = \left(1 + cos(\frac{p}{q}t)\right) \times cos(t) \\ y(t) = \left(1 + cos(\frac{p}{q}t)\right) \times sin(t) \end{cases}$ pour p et q entiers positifs et $t \in [0, 2\pi]$.

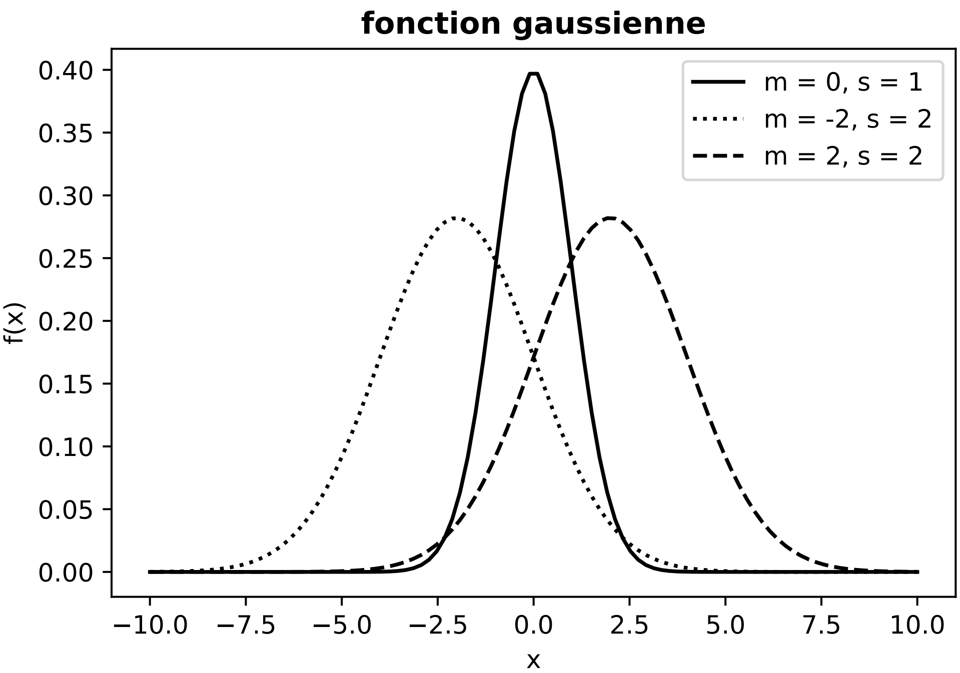
Exercice 3 : Implémenter une fonction gaussienne

Q1. Créer la fonction : `gauss(x, m = 0, s = 1)`, qui modélise la gaussienne :

$$f(x) = \frac{1}{\sqrt{2\pi} s} \exp\left[-\frac{1}{2}\left(\frac{x-m}{s}\right)^2\right]$$

Q2. Créer un tableau `x` à l'aide de la fonction `linspace`, du module `numpy`, pour `100` valeurs `x` uniformément espacées dans [-10, 10].

Q3. Écrire les instructions pour tracer le graphique ci-dessous à l'aide de la bibliothèque `matplotlib`.



Exercice 4 : La suite logistique

Soit r un réel de $[0, 4]$. On considère la suite logistique définie par

$$u_{n+1} = r \times u_n(1 - u_n) \quad et\ u_0 = 0.5$$

. Cette suite très célèbre permet de modéliser certaines dynamiques de population. Elle a un comportement très différent selon les valeurs de r. On se propose de tracer le nuage de points suivants.

Q1. Écrire une fonction `Liste_suite(u0,r,n)` qui renvoie la liste des n premiers termes de la suite.

Q2. Créer une subdivision uniforme `R` de l'intervalle $[0,4]$ en 101 points.

Q3. Pour chaque élément r de `R`, représenter les points de coordonnées $(r,u_{100}),(r,u_{101}),\dots,(r,u_{200})$.