Informatique tronc commun – TP 05

16 novembre 2016

1. **Lisez attentivement tout l'énoncé avant de commencer.**
2. Commencez la séance en créant un dossier au nom du TP dans le répertoire dédié à l'informatique de votre compte.
3. Après la séance, vous devez rédiger un compte-rendu de TP et l'envoyer au format électronique à votre enseignant.
4. Vous rendrez un compte-rendu pour chaque séance sous forme d'un fichier d'extension `.py`, en respectant exactement les spécifications données plus bas.
5. Ce TP est à faire en binôme, vous ne rendrez donc qu'un compte-rendu pour deux.
6. Ayez toujours un crayon et un papier sous la main. Quand vous réfléchissez à une question, utilisez les !
7. Vous devez être autonome. Ainsi, avant de poser une question à l'enseignant, merci de commencer par :
   — relire l'énoncé du TP (beaucoup de réponses se trouvent dedans) ;
   — relire les passages du cours [1] relatifs à votre problème ;
   — effectuer une recherche dans l'aide disponible sur votre ordinateur (ou sur internet) concernant votre question.

   Il est alors raisonnable d'appeler votre enseignant pour lui demander des explications ou une confirmation !

Le but de ce TP est d'apprendre à représenter des fonctions avec Python, en prenant comme exemple l'approximation d'un signal par les séries de Fourier (un cadre déjà vu en physique).

Instructions de rendu

Attention : suivez précisément ces instructions. Votre fichier portera un nom du type `tpXX_berne_zannad.py`, où `XX` est à remplacer par le numéro du TP et les noms de vos enseignants par ceux des membres du binôme. Le nom de ce fichier ne devra comporter ni espace, ni accent, ni apostrophe. Dans ce fichier, vous respecterez les consignes suivantes.

1. Dans le cas fort improbable où vous ne vous en souviendriez pas.

— Écrivez d'abord en commentaires (ligne débutant par #), le titre du TP, les noms et prénoms des étudiants du groupe.
— Commencez chaque question par son numéro écrit en commentaires.
— Les questions demandant une réponse écrite seront rédigées en commentaires.
— Les questions demandant une réponse sous forme de fonction ou de script respecterons pointilleusement les noms de variables et de fonctions demandés.

1 Tracé d'une fonction simple

En utilisant Python, on peut tracer de nombreux types de graphiques. Nous allons utiliser une bibliothèque regroupant de (très) nombreuses fonctions de tracé : `matplotlib`. En fait, cette bibliothèque est bien trop vaste, nous n'utiliserons que sa sous-bibliothèque `matplotlib.pyplot`.

Télécharger le script `tp05_ex_sin.py` sur le site de la classe, l'enregistrer dans le répertoire dédié au TP puis l'ouvrir avec IDLE. Éxécuter le script après l'avoir lu.

Q1 Quel est le type de `x` ?

Q2 Comment Python représente-t-il graphiquement une fonction ?

Q3 Où le tracé s'arrête-t-il ? Pourquoi ?

On pourra ensuite consulter le script `tp05_ex_complet.py` sur le site de la classe. Il présente un tracé de fonctions plus complet.

On peut simplement créer une liste d'abscisses en utilisant la fonction `linspace` de la bibliothèque de calcul `numpy`. Charger cette fonction dans l'interpréteur interactif par la commande

```
from numpy import linspace
```

puis consulter son manuel par la commande `help(linspace)`.

Q4 Donner la commande permettant de définir `x` de manière plus appropriée dans l'exemple `tp05_ex_sin.py`.

Q5 Écrire une fonction `transitoire(A,nom_de_fichier)` qui enregistre dans le fichier `nom_de_fichier` le graphe des fonctions

$$t \mapsto A\left(1 - \mathrm{e}^{-t/\tau}\right)$$

sur $[0, 10]$, pour chaque $\tau \in \left\{\frac{1}{2}; 1; 2; 4; 8\right\}$. *Attention au type de `nom_de_fichier` !*

2 Synthèse de Fourier

On s'intéresse maintenant à l'approximation d'un signal périodique par des fonctions trigonométriques. On considère sur $\mathbb{R}$ la fonction créneau, impaire et périodique de période 2, définie par $C(1) = 0$ et sur $]0, 1[$ par

$$C : t \mapsto 1.$$

Soit aussi la fonction triangle, définie sur $\mathbb{R}$, paire et périodique de période 2, définie sur $[0,1]$ par

$$T : t \mapsto 1 - 2t.$$

On peut montrer que, en tout réel t où C est continue,

$$C(t) = \frac{4}{\pi} \sum_{p=0}^{+\infty} \frac{1}{2p+1} \sin((2p+1)\pi t).$$

De même, pour tout $t \in \mathbb{R}$,

$$T(t) = \frac{8}{\pi^2} \sum_{p=0}^{+\infty} \frac{1}{(2p+1)^2} \cos((2p+1)\pi t).$$

Ce symbole $\displaystyle\sum_{p=0}^{+\infty}$ doit être vu comme une limite et sera défini dans le cours de mathématiques. On approche alors, pour tout $t \in \mathbb{R}$, $C(t)$ et $T(t)$ respectivement par les sommes

$$C_n(t) = \frac{4}{\pi} \sum_{p=0}^{n} \frac{1}{2p+1} \sin((2p+1)\pi t) \text{ et } T_n(t) = \frac{8}{\pi^2} \sum_{p=0}^{n} \frac{1}{(2p+1)^2} \cos((2p+1)\pi t).$$

Les physiciens appellent le terme en $p = 0$ le *fondamental* et les autres termes les *harmoniques*.

Q6 Écrire une fonction `creneau(t)` retournant la valeur de $C(\mathtt{t})$ (pour `t` réel). *Indice : on pourra considérer la parité de la partie entière de* `t`.

Q7 Écrire une fonction `sp_creneau(n,t)` retournant la valeur de $C_{\mathtt{n}}(\mathtt{t})$ (pour `n` entier et `t` réel).

Q8 Écrire une fonction `fourier_creneau(nom_de_fichier)` ne retournant rien et enregistrant dans `nom_de_fichier` le graphe sur l'intervalle $[0,4]$ de C, celui de son fondamental (*i.e*, C_0), ainsi que ceux de C_3, C_5 et C_{100}.

Pour les plus rapides, voici des questions supplémentaires.

Q9 Écrire une fonction `triangle(t)` retournant la valeur de $T(\mathtt{t})$ (pour `t` réel).

Q10 Écrire une fonction `sp_triangle(n,t)` retournant la valeur de $T_{\mathtt{n}}(\mathtt{t})$ (pour `n` entier et `t` réel).

Q11 Écrire une fonction `fourier_triangle(nom_de_fichier)` ne retournant rien et enregistrant dans `nom_de_fichier` le graphe sur l'intervalle $[0,4]$ de T, celui de son fondamental (*i.e*, T_0), ainsi que ceux de T_3, T_5 et T_{100}.