

Licence L2
Programmation orientée objet
(Introduction au Python) : **EI31T**

Durée: 2H. Documents autorisés : notes personnelles.

Chaque candidat doit, au début de l'épreuve, porter son nom dans le coin de la copie qu'il cachera par collage après avoir été pointé. Il devra en outre porter son numéro de place sur chacune des copies, intercalaires, ou pièces annexées.

1. Écrire en Python une fonction qui renverse une liste *sur place*, c'est-à-dire, qui retourne la même liste, mais avec l'ordre des éléments renversé :

a	b	c	d	e	f
---	---	---	---	---	---

 →

f	e	d	c	b	a
---	---	---	---	---	---

Il faut déplacer les éléments un par un dans la liste (tableau), vous n'avez pas le droit (bien sûr...) d'utiliser des fonctions ni méthodes prédéfinies que renversent les listes, ni copier les éléments dans une nouvelle liste. Le programme doit fonctionner avec des listes de longueur quelconque, contenant éléments arbitraires.

2. Écrire un programme qui demande à l'utilisateur un nombre entier positif, disons N , et ensuite entre dans une boucle **for** qui calcule la somme $s = \sum_{k=1}^M (-1)^k \frac{1}{k^2}$. La borne supérieure M est nominale égale à N , mais si la valeur (absolue) suivante du terme ajouté est inférieure à ϵ , disons, $\epsilon = 0.00001$ (mais ceci doit être considéré comme variable, non pas une constante permettant de précalculer M), la boucle doit se terminer de manière précoce. Le résultat doit être affiché. Penser à utiliser un itérateur standard, prédéfini, qui permet de parcourir la liste $[1,2,3,4, \dots]$ sans créer une liste très longue.
3. Définir une classe d'objets, disons **s1** qui hérite **list**. L'initialiseur **__init__** prend un paramètre spécifique (en addition à **self**) – une liste dont la structure est la suivante, c'est une liste de listes, par exemple

```
a = s1([[1,2,11,3],[4,6,9,10],[5,0,9,8]])
```

et attribue ce paramètre à une variable interne. Construire la méthode **__repr__** qui simplement affiche cette variable. Construire la méthode qui permet l'adressage des éléments de la structure, comme si c'était un tableau rectangulaire (ci dessus : 3 lignes de quatre colonnes), par la séquence contenant deux valeurs : ligne et colonne, par exemple **a[3,1]**. C'est tout, ne compliquez pas des choses, le programme doit avoir moins de 10 lignes de code. **Bonus supplémentaire.** Comment étendre la méthode aux objets représentant des listes des listes ... des listes, de profondeur quelconque, et dont l'adressage des éléments devrait donc **a[i,j,k,...,m]** ?

Rappel : La structure syntaxique **i, j, k** est un objet de type séquence, et peut être indexée.

Attention : Je vous prie d'écrire de manière lisible, tout texte difficile à déchiffrer risque d'être rejeté. Ne jamais copiez des morceaux de code inutile, tout code hors sujet (surtout : tout le code des notes du cours, etc.) sera pénalisé ! Évitez toute discussion qualitative (« philosophique ») inutile. Par contre, commentez votre code, sinon il risque d'être peu compréhensible.