

TP 2 : conditionnement et boucles

Instructions conditionnelles

La syntaxe des instructions conditionnelles en Python est la suivante :

```
if condition 1:
    Instructions 1
elif condition 2:
    Instruction 2
else:
    Instructions 3.
```

Boucles

Répétitions inconditionnelles

Un intervalle d'entiers s'exprime par `range` en Python. Précisément, `range(a,b)` génère tous les entiers k vérifiant $a \leq k < b$. On peut également ajouter un "pas" entre deux entiers successifs.

```
for compteur in range(a,b):
    Instructions 1
```

On peut également utiliser autre chose qu'un intervalle d'entier (une liste par exemple).

Répétitions conditionnelles

```
Initialisation de la variable utilisée dans le test
while condition:
    Instructions
```

Attention : si la condition testée ne change pas et ne devient jamais fausse, le bloc d'instruction qui suit `while` est répété indéfiniment et le programme ne se termine jamais. Pour l'interrompre manuellement, on appuie simultanément sur les touches `Ctrl` et `C`.

Exercices

Écrire, sans utiliser les fonctions correspondantes de Python, dans chacun des cas suivants, un programme ou une fonction qui :

1. à un nombre x associe sa valeur absolue.
2. à un réel x associe sa partie entière.
3. à deux entiers a et b associe le reste de la division euclidienne de a par b .
4. à deux réels x et y associe le plus grand des deux.
5. à une année associe le fait qu'elle soit bissextile ou non.
6. à un entier n associe le terme de rang n de la suite définie par $u_0 = 1$ et $\forall k \geq 0, u_{k+1} = 2u_k - 1$.
7. à un entier n associe la somme des n premiers termes de la suite précédente.
8. à un entier n associe sa factorielle.
9. à un réel a et un entier n associe a^n .
10. à un entier n associe $\sum_{1 \leq i \leq j \leq n} \frac{i}{j}$.
11. à deux nombres n et p associe $\binom{n}{p}$.
12. à un entier n associe les n premières lignes du triangle de Pascal sous forme de texte.