



Chapitre 3

Codages des nombres en Python

TP

Simon Dauguet
simon.dauguet@gmail.com

15 octobre 2025

Exercice 1 :

Pour tout entier naturel n , on définit $u_n = \frac{(1+10^{-n})-1}{10^{-n}}$.

Définir un code  qui affiche les valeurs de u_n pour $n \leq 20$. Expliquer les résultats.

Exercice 2 :

On souhaite calculer le nombre dérivé en 1 de la fonction $t \mapsto t^2$.

- Proposer une fonction `nbDerive1(h: float) -> float`, qui renvoie $((1+h)^2 - 1)/h$.
- Simplifier mathématiquement l'expression $((1+h)^2 - 1)/h$ et écrire une autre fonction qui renvoie le calcul.
- Préciser les deux résultats pour $h \in \{10^0, 10^{-1}, \dots, 10^{-20}\}$. Expliquer les phénomènes.

Exercice 3 (Monayeur) :

Un monayeur est un système informatique permettant de calculer un rendu de monnaie. On propose le code suivant :

```
1 def monayeur(prix:float, monnaie:float) -> float :
2     assert prix < monnaie , "Pas assez de monnaie"
3     Banque = [20,10,5,2,1,0.5,0.2,0.1,0.05,0.01]
4     rendu = [0]*len(Banque)
5     diff = monnaie-prix
6     for k in range(len(Banque)) :
7         rendu[k] = int(diff/Banque[k])
8         diff = diff-rendu[k]*Banque[k]
9     return(rendu)
```

- Que fait cet algorithme ?
- Quel est le sens de la ligne 7 ? Et de la ligne 8 ?
- Recopier cet algorithme et tester `monayeur(12,28)`. Vérifier ce que renvoie l'algorithme.
- Tester `monayeur(4.2,12.7)` et faire une vérification. Commenter.
- Tester `monayeur(4.1,12.7)` et faire une vérification. Commenter.
- Proposer une correction de l'algorithme en expliquant vos choix.