

#### Séance 4 : Boucle bornée (for) : quand on connaît le nombre de répétitions

Définition : Une boucle permet de répéter un ensemble d'instructions.

La boucle « Pour » s'utilise avec un compteur qui a une valeur de départ et une valeur de fin déterminée et qui s'incrémente (de 1 par défaut) à chaque tour de la boucle.

En Python : On a : **for i in range ( début , fin, pas ) :**  
*instructions*

Le mot clé **for** introduit la boucle. Les deux-points **:** à la fin de la ligne ouvrent le bloc indenté (décalé vers la droite) comportant les instructions à répéter.

C'est le retour vers la gauche, à la ligne suivante, qui marque la sortie de la boucle (FinPour).

La variable « compteur » de notre boucle dans l'exemple ci-dessus est **i**.

Pour la première exécution de la boucle (partie *instructions*), elle prend la valeur *début*.

Après la première exécution de la boucle, si sa valeur est strictement inférieure à *fin*, elle prend la valeur *début + pas*, et une deuxième exécution de la boucle est faite.

Et on recommence, ainsi de suite, jusqu'à ce que la valeur de **i** soit supérieure ou égale à *fin*.

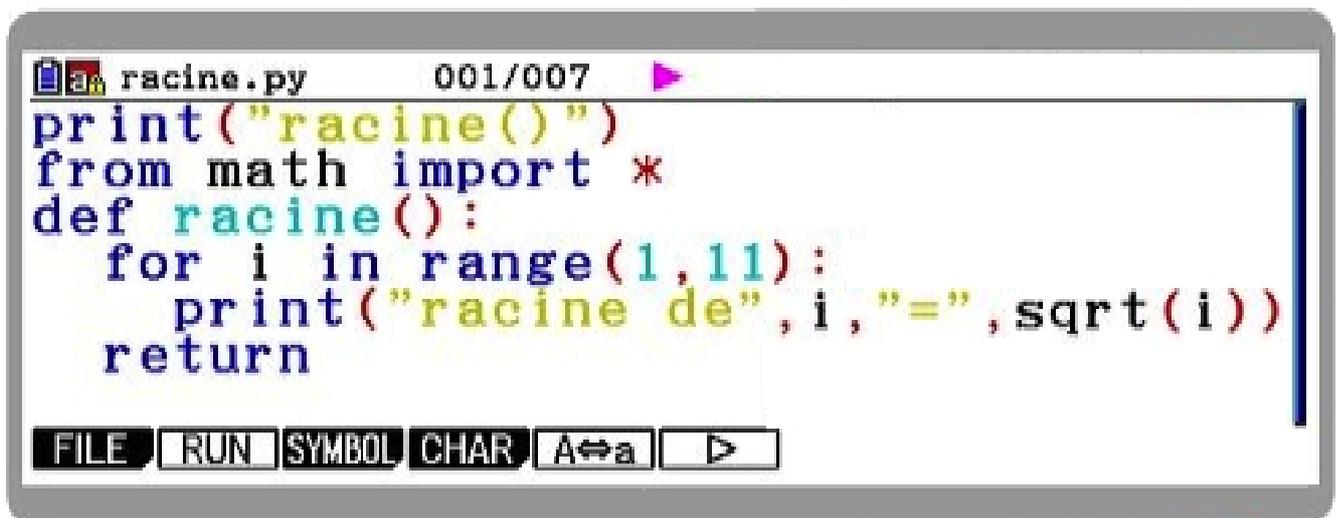
Alors la boucle s'arrête.

*Remarque* : si *pas* n'est pas précisé, alors, par défaut, la valeur du compteur est augmentée de 1 à chaque fois.

Exemple : Écrivons un script, *racine*, qui affiche la racine carré de tous les entiers de 1 à 10.

On prendra 1 comme valeur de *début* et 10 + 1 comme valeur de *fin* (pour s'arrêter à 10).

**N. B.** : il faudra importer la bibliothèque « math » pour pouvoir utiliser racine carrée  $\sqrt{x}$ .



```
racine.py 001/007 ▶
print("racine()")
from math import *
def racine():
    for i in range(1, 11):
        print("racine de", i, "=", sqrt(i))
    return
```

Comment modifier ce script pour que l'on puisse choisir la valeur de départ ? de fin ?

→ **Coup de pouce vidéo** : <https://www.youtube.com/watch?v=17ldIQqAToI>  
et rappels dans le livre page 346

Application : Écrire un script, *table*, sur ta calculatrice, utilisant une fonction et qui permet d'afficher les *max* premiers nombres de la table de *tabl*.

```
table.py 001/006 ▶  
print("table(tabl,max)")  
def table(tabl,max):  
    for i in range(1,max+1):  
        print(i,"x",tabl,"=",i*tabl)  
    return
```

FILE RUN SYMBOL CHAR A⇌a ▶