

# Formation Module Extension Python c/c++ Partie 1

Ihab ABADI / UTOPIOS

# Sommaire

- Nécessité d'extension de Python par c/c++.
- Les modules pour l'interaction de Python avec d'autre langages.
  - Python C API
  - Cython
  - Ctypes
- Types python vs c/c++
- Utilisation des méthodes avec args
- Utilisation des pointeurs.
- Utilisation des structures

# Nécessité d'extension de Python par c/c++.

- Pour étendre les fonctionnalités de Python
- Pour améliorer les performances
- Pour utiliser Python comme langage de collage
- Pour créer des liaisons Python pour une bibliothèque

# Python C API

- Python C API est une librairie qui permet de développer des extensions Python En C ou C++
- Python-dev fournit des structures C pour exécuter des fonctionnalités C.
- Nécessite des connaissances de développement en C.
- Un exemple.

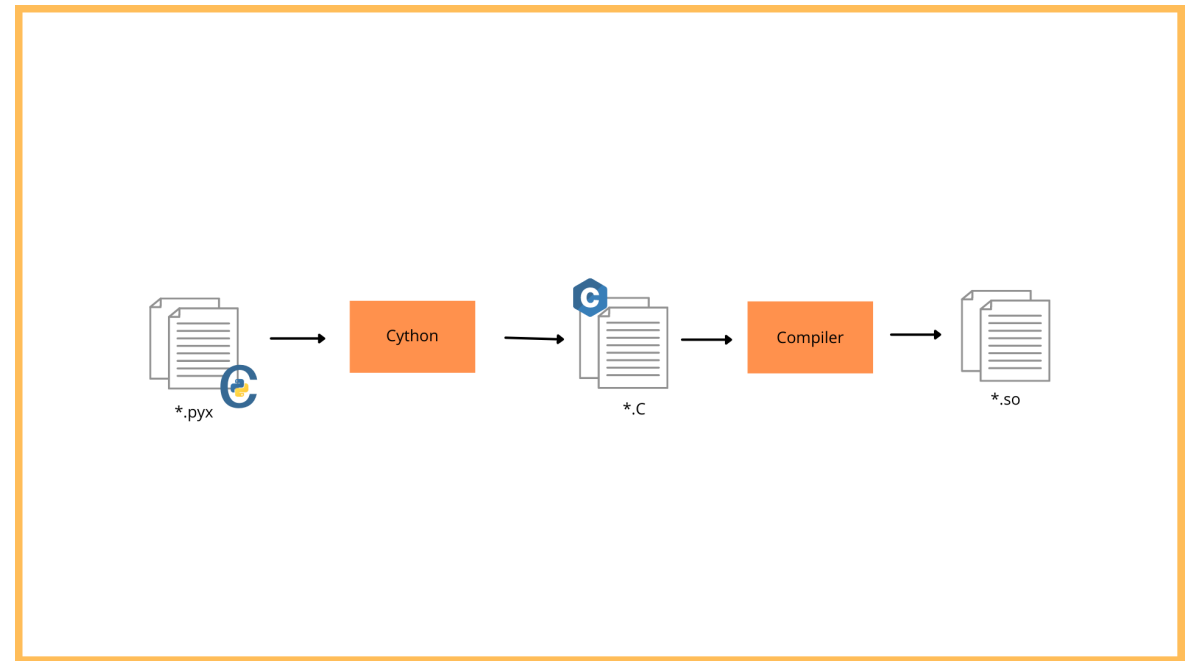
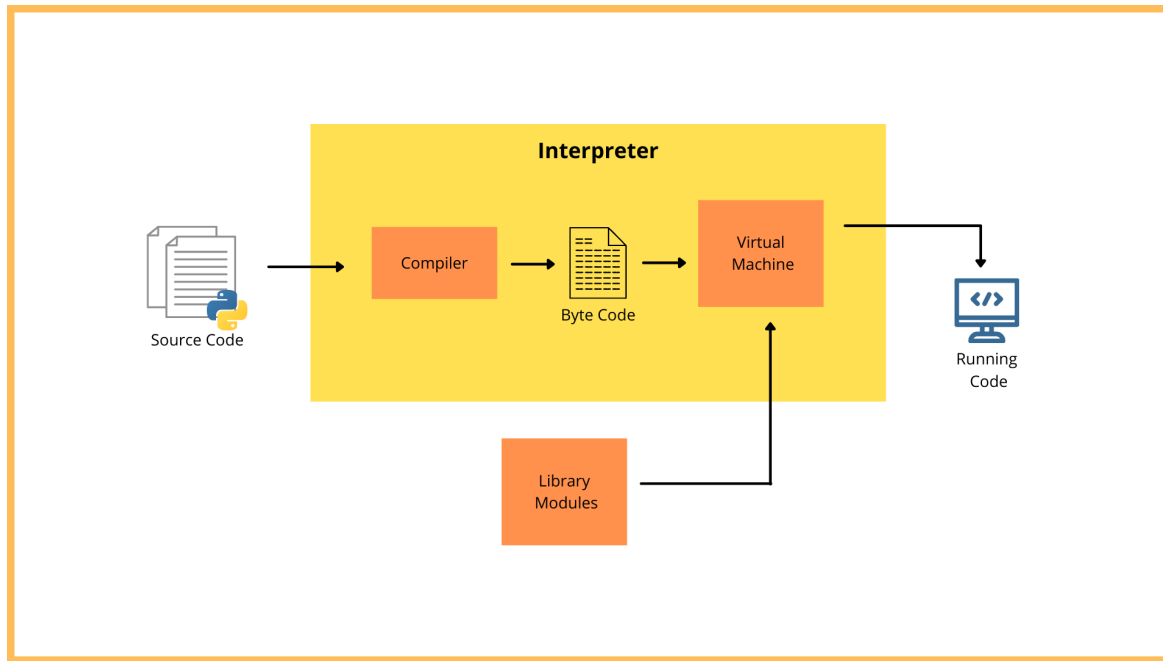
# Cython

- Cython peut être considéré à la fois comme un module et comme un langage de programmation qui étend (en quelque sorte) Python en permettant l'utilisation du typage statique emprunté à C/C++.
- Tout le code Python est valide Cython, mais pas l'inverse.
- Vous pouvez directement copier votre code Python existant dans un fichier Cython, puis le compiler pour améliorer les performances.

# Cython

- Python est plus efficace que C étant donné qu'il s'agit d'un langage de haut niveau.
- Python est efficace mais lent. C, en revanche, est moins efficace mais plus rapide que Python.
- Cython vise donc à apporter tous les avantages de C à Python tout en maintenant l'efficacité de Python.

# Cython



# Cython

- Utiliser Cython partout ne garantit pas toujours une vitesse accrue.
- vous pouvez l'utiliser dans des fonctions qui impliquent un grand nombre d'opérations mathématiques et d'itérations de boucle..
- Un autre cas d'utilisation intéressant est lorsque vous disposez déjà d'une bibliothèque C ou C++ nécessitant une interface Python. Dans ce cas, vous pouvez utiliser Cython pour créer un wrapper pour la bibliothèque.



# Exercice Cython

Exercice

- Créer une fonction en Cython pour calculer La suite de Fibonacci et comparer le temps d'exécution à une fonction en Python

# Exercice Cython

Exercice

- Créer une fonction en Cython pour trouver les nombres premiers et comparer le temps d'exécution à une fonction en Python

# Cython

- Cython nous permet également d'appeler des fonctions ou librairies c/c++ dans un script python.
- Démo

# Exercice Cython

## Exercice

- Créer un script cython qui permet d'exécuter la fonction c suivante:

```
#include <stdio.h>
#include <dirent.h>

void getDirectory(const char* folder) {
    DIR *d;
    struct dirent *dir;
    d = opendir(folder);
    if (d) {
        while ((dir = readdir(d)) != NULL) {
            printf("%s\n", dir->d_name);
        }
        closedir(d);
    }
}
```