# Programmation par composants

# Exercice 1 diapo 27 :

Premièrement, nous avons cloné le référentiel Github à partir de l’URL fournie : <https://github.com/jluuM2/python_cpp_2023> avec la commande suivante :

* git clone <https://github.com/jluuM2/python_cpp_2023.git>

Une image contenant texte, capture d’écran, Police, noir

Description générée automatiquement

Ensuite, nous sommes allés dans le répertoire python\_cpp\_2023, avons utilisé les commandes git submodule init et git submodule update et enfin, nous sommes allés dans le répertoire hello.

Une image contenant texte, capture d’écran, Police, noir

Description générée automatiquement

Ensuite, nous avons fait make pour compiler : Une image contenant texte, Police, capture d’écran

Description générée automatiquement

# Exercice 2 diapo 28 :

Nous écrivons le fichier test\_cle.py :

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Il est présent dans le répertoire python\_cle\_publique\_2023 :

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Nous écrivons aussi composant\_cle.py dans ce même répertoire :

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Nous exécutons test\_cle.py avec ‘python3 test\_cle.py’ mais nous n’avons pas les accès à micro-ecc :

Une image contenant texte, capture d’écran, Police, noir

Description générée automatiquement

C’est pourquoi nous changeons de stratégie et importons ecdsa dans composant\_cle.py :

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Et voici test\_cle.py : Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Cela nous donne quand on exécute :

