
TP 2 : premiers programmes

Tout exercice non marqué d'un ♣ est à terminer pour la semaine prochaine.

Exercice 1 (Notes de cours).

- (1) Consultez le poly de l'Amphi 1 sur la page web du cours.
- (2) Que signifie *syntaxe*? Et *sémantique*? Vérifiez dans le poly.
- (3) Quelle est la syntaxe et la sémantique de l'affectation de variables? Vérifiez dans le poly.

Exercice 2 (Téléchargement du sujet de TP).

- (1) Ouvrez un terminal (vous pouvez consulter l'exercice 2 du TP 1).
- (2) Vérifiez si vous avez déjà un dossier `ProgImperative` :

```
ls ~/ProgImperative
```

Si ce n'est pas le cas (par exemple parce que vous avez utilisé un compte de secours la semaine dernière), créez le avec :

```
mkdir ~/ProgImperative
```

- (3) Exécutez la commande `info-111` et observez le résultat.
- (4) Vous venez de lire la documentation de la commande `info-111`. Téléchargez le sujet de TP « Semaine2 » en suivant les instructions données à ce sujet par cette documentation.
- (5) Observez ce qui est affiché ; notez que `info-111` a fait appel à la commande `git clone . . .`.
- (6) Consultez le nouveau contenu de votre répertoire `ProgImperative` ainsi que celui de ses sous-dossiers.

Exercice 3 (Jupyter : affectations et conditionnelles).

- (1) Démarrez Jupyter :
 - **En salle de TP** : vous n'avez pas besoin de vous connecter au serveur JupyterHub. Si vous ne vous souvenez plus de la commande exacte, tapez `info-111` dans un terminal.
 - **Sur JupyterHub** : Jupyter est déjà lancé!
- (2) Naviguez jusqu'à votre répertoire `Semaine2`.
- (3) Sélectionnez la feuille de travail `feuille1-valeurs-types-variables.ipynb`. Suivez les instructions qu'elle contient. Rappel : ne fermez pas le terminal où vous avez le cas échéant lancé Jupyter.
- (4) De même avec `feuille2-conditions-fonctions.ipynb`.

Exercice 4 (Soumission de votre travail).

- (1) Ouvrir un nouveau terminal (voir TP 1).
- (2) Exécutez la commande `info-111`
- (3) Suivez les instructions indiquées pour soumettre votre travail, en veillant à bien préciser votre groupe.
- (4) N'oubliez pas de soumettre à nouveau votre travail en fin de séance.

Exercice 5 (Serveur JupyterHub).

Rappel : l'université a déployé un service **JupyterHub** (expérimental) vous permettant de travailler en dehors des séances avec simplement une connection internet et un navigateur web (firefox, chrome, safari, ...).

Cet exercice n'est à faire que si vous n'avez pas encore utilisé ce service ou si vous avez rencontré des difficultés.

- (1) Suivez les instructions du TP 1 pour vous connecter au service JupyterHub, y télécharger vos TP 1 et 2 et les consulter ; demandez de l'aide au besoin à votre enseignant.
- (2) Fermez votre serveur avec **Control Panel** -> **Stop My Server**, puis fermez la page.

Exercice 6 (Retour sur le TD).

- (1) Ouvrez la feuille `feuille-3-td-exercice-3` et implantez en C++ les algorithmes de l'exercice 3 du TD.
- (2) De même pour l'exercice 4 dans la feuille `feuille-4-td-exercice-4`.

Exercice 7 (**Pour la semaine prochaine**).

- (1) Vérifiez que vous pouvez vous connecter au serveur **JupyterHub** depuis chez vous et que vous pouvez y exécuter vos fiches Jupyter (voir exercice 5).
- (2) Terminez les fiches Jupyter de ce TP, soit en salle informatique de l'université, soit depuis chez vous en utilisant le serveur **JupyterHub**.

Exercice ♣ 8 (Pour aller plus loin).

Refaites le TP en utilisant cette fois le langage **Python**. Pour cela, lancez **Jupyter** comme précédemment, puis choisissez **New** -> **Python 3**. Cela créera une nouvelle feuille de travail où vous pourrez écrire du code Python.

Vous trouverez un cours complet sur Python à cette adresse :

<http://fr.openclassrooms.com/informatique/cours/apprenez-a-programmer-en-python>

Il est intéressant de le lire et de comparer avec ce que vous apprenez en C++ ce semestre.

Pour que la soumission tienne compte de vos nouveaux fichiers, vous devez au préalable les déclarer au système de gestion de version sous-jacent **git** ; par exemple, pour déclarer un nouveau fichier `exercice8.ipynb`, vous ferez :

```
cd ~/ProgImperative/Semaine2
git add exercice8.ipynb
```

Il sera alors inclus dans toutes les soumissions ultérieures.
(voir suite au dos)

Un peu de syntaxe Python :

Une fonction :

```
def maFonction(argument):  
    instructions
```

Une conditionnelle :

```
if a == 1:  
    ...  
else:  
    ...
```

Une boucle :

```
while a <= 10:  
    ...  
    a += 1
```

Exercice ♣ 9 (Euler forever!).

Vous trouverez sur le site du projet Euler <https://projecteuler.net> une série de problèmes mathématiques qui nécessitent chacun une combinaison de réflexion sur feuille et de programmation (voir <http://submoon.freeshell.org/fr/sphinx/euler.html> pour avoir les énoncés en français).

Essayez de résoudre les problèmes 1, 2, 5 et tous ceux qui vous plairont !