Séance 1: Premiers Pas

L1 – Université Nice Sophia Antipolis

Objectifs:

- Apprendre à manipuler le toplevel de Idle
- Utiliser Python comme une calculette évoluée
- Savoir écrire et exécuter un programme avec Idle
- Introduction aux entrées-sorties dans un terminal

Travail sous Windows au 2ème étage (CRIPS)

Vous allez travailler sur des ordinateurs sous le système d'exploitation Windows-8. Peut-être utilisez-vous comme la plupart de vos enseignants un ordinateur sous UNIX : Linux, ou MacOS-X ? Aucune importance puisque les logiciels qui nous serviront à programmer en Python 3 fonctionnent sur tous ces systèmes quasiment à l'identique. Nous supposons que vous savez utiliser votre ordinateur pour copier/déplacer des fichiers et lancer un logiciel ou un navigateur Web (Firefox, Chrome, etc).

Vous êtes donc devant votre machine. Vos notes de cours sont à portée de main et vos neurones commencent leurs échanges électro-chimiques, bravo !

Compte ENT

Vérifiez que vous savez accéder à votre compte ENT, lire vos mails, et utiliser OneDrive.

Stockage des fichiers

Nous vous recommandons d'acheter une clé USB (à partir de 5 euros). Certaines clés USB font aussi office de lecteurs MP3 ou de téléphones portables.

- si vous n'avez pas de clé USB : votre **espace de travail** durant le TP sera le *bureau* de Windows. A la fin du TP, sauvez vos fichiers sur OneDrive, **sinon ils seront perdus**.
- si vous avez une clé USB : c'est plus simple, travaillez directement sur votre clé USB qui sera votre disque dur et votre **espace de travail** durant ce TP.

Avant de commencer ce TP : sur OneDrive ou sur votre clé USB, créez un répertoire `Python` dans lequel vous rangerez tous vos fichiers de TP.

Lancement de Python 3

Dans le menu de Windows-8, cherchez `Python 3 64 bits`. Attention, il y a aussi un vieux Python 2 installé sur les machines, vérifiez bien qu'il s'agit de **Python 3**...

Lisibilité

Il est de bonne pratique de choisir une police de taille fixe (Courier ou Monaco) et suffisante (16 par exemple) pour ne pas fatiguer les yeux, d'avoir un fond d'écran non blanc (jaune pâle par exemple), et de laisser un espace de part et d'autre d'un opérateur : il est plus facile de lire `2 - x + 3 // y` que `2-x+3//y` si l'on est à 1 mètre d'un écran !

Exercice 1 (Premiers pas dans le toplevel, ⋆)

1. *Définissez deux variables :* `p` *ayant pour valeur 5 et* `q` *ayant pour valeur 3p.*
2. *Avec une seule instruction print(. . .) et en utilisant les variables p et q, faites afficher la phrase suivante :* `p vaut 5 et q vaut 15 , leur somme fait 20`.
   N.B *La solution suivante n'est* pas *celle que l'on attend bien entendu :* `print('p vaut 5 et q vaut 15, leur somme fait 20')`
3. *Demander en python «* `q` est-il un multiple de `p` *? ». Indication : utilisez le reste de la division.*
4. *Au toplevel, traduisez en Python la phrase suivante :*

   « si `q` est un multiple de `p`, afficher `OUI` sinon afficher `NON` *»*
5. *Sans consulter vos notes de cours, écrivez au toplevel les lignes de code permettant d'échanger les valeurs de* `p` *et* `q`*, en utilisant une variable temporaire* `tmp`*. Vérifiez ensuite que les valeurs de* `p` *et* `q` *ont bien été permutées.*
6. *Calculez la moyenne des entiers de 1 à 10. Le résultat est un nombre flottant (à virgule).*
7. *Calculez la moyenne des entiers de 1 à 11. Ne retaper pas la formule complète, mais modifiez la précédente en tapant* ***Alt-p*** *(****Ctl-p*** *sur Mac) qui va chercher les lignes précédemment entrées (l'*historique*). On utilise* ***Alt-n*** *(****Ctl-n*** *sur Mac) en sens inverse.*
8. *Comment feriez-vous pour savoir si la fraction* $51/85$ *est irréductible ? En d'autre termes, peut-on la simplifier ? Par combien ?*
9. *Demandez en une ligne si* 5^4 *est plus grand que* 4^5
10. *Tapez au toplevel :* `help(max)` *pour demander une petite doc de la fonction* `max`

□

Du toplevel aux vrais programmes

Un vrai programme est une suite d'instruction lue et exécutée d'un seul bloc et enregistré dans un fichier. Vous allez écrire votre premier programme dans l'éditeur de Idle. Dans le menu, choisissez **File/New File**, puis tapez votre programme dans la nouvelle fenêtre, et quand vous êtes prêts, *exécutez* votre programme en choisissant **Run/Run Module**.

Lorsque l'on vous demandera de sauver le contenu de votre éditeur, vous opterez pour un fichier de nom tp1.py sur votre espace de travail, dans un dossier nommé Python.

Exercice 2 (Pierre-Feuille-Ciseaux, ⋆⋆)

Le jeu pierre-feuille-ciseaux (aussi appelé parfois mourre *en Italie ou* 手勢令 *en Chine) est un jeu de mains où deux joueurs s'affrontent en choisissant simultanément chacun un objet parmi pierre, feuille, ou ciseaux. Si les deux joueurs choisissent le même objet, il y a égalité, sinon la feuille l'emporte sur la pierre, qui l'emporte sur les ciseaux, qui l'emportent sur la feuille. Vous allez maintenant programmer un jeu de mourre pour affronter votre ordinateur !*

1. *Écrivez un programme qui affiche la phrase «* `quel est ton choix, humain?` *» puis qui demande à l'utilisateur de saisir un texte, et enfin affiche la phrase «* `tu as choisi XXXX, humain` *», où* `XXXX` *est remplacé par le texte saisi par l'utilisateur juste avant. Par exemple :*

```
quel est ton choix, humain? pierre
tu as choisi pierre, humain

```

   Vous utiliserez la fonction `input(msg)` *qui affiche le message contenu dans le paramètre* `msg` *et renvoie ce que l'utilisateur a saisi au clavier à la suite de ce message.*
2. *Modifiez votre programme pour qu'il affiche la phrase «* `ok` *» si l'utilisateur a bien saisi l'un des trois mots* pierre, feuille, *ou* ciseaux, *et sinon affiche la phrase «* `choix incorrect, je choisis pour toi` *». Par exemple, on aura :*

```
    quel est ton choix, humain? pierre
    ok

```

ou encore

```
    quel est ton choix, humain? puits
    choix incorrect, je choisis pour toi ...

```

3. *Revenez au début du programme et définissez une fonction* `choix_ordi()` *qui renvoie un mot au hasard parmi les trois mots possibles (vous aurez besoin de la fonction* `randint` *du module* `random` *vue en cours). Complétez votre programme pour que le message d'erreur précédant indique le choix qu'a fait l'ordinateur. On aura désormais*

```
    quel est ton choix, humain? puits
    choix incorrect, je choisis pour toi : feuille

```

4. *Complétez votre programme pour faire une partie complète : l'ordinateur fait un choix pour lui au début (mais le garde secret), puis l'utilisateur fait un choix, puis le programme affiche les deux choix et le gagnant éventuel. Par exemple :*

```
    quel est ton choix, humain? ciseaux
    ok
    humain : ciseaux
    ordi : feuille
    gagnant : humain

```

□

Exercice 3 (Le juste prix, ⋆⋆⋆)

Le juste prix consiste à deviner le prix d'un objet (un entier rond, inférieur ou égal à 100 euros) en faisant des propositions de prix, l'autre joueur indiquant si on est au-dessus ou en-dessous du prix réel.

1. *En vous inspirant de l'exercice précédant, programmez le jeu du juste prix où c'est l'humain qui doit deviner le prix. Il a droit à 7 propositions. Indication :* `int('42') == 42`.
2. *Quelle stratégie adoptez-vous pour gagner à coup sûr ?*
3. *Programmez le jeu où c'est l'ordinateur qui fait des propositions et l'humain qui répond «* au-dessus *» ou «* au-dessous *». L'ordinateur a droit à 7 propositions et devra gagner à coup sûr, ou afficher un message spécial si l'humain a triché dans ses réponses.*

□