

TP3 – Introduction aux boucles et expressions conditionnelles

Exercice 1 : Utilisation de la boucle FOR et de l'expression conditionnelle IF

Construisez une boucle qui parcourt les nombres de 0 à 20 et qui affiche une liste contenant les nombres pairs inférieurs ou égaux à 10 d'une part, et une liste contenant les nombres impairs strictement supérieurs à 10 d'autre part.

Pour cet exercice,

- vous pouvez initialiser vos listes en utilisant des listes vides :

```
nb_pairs_inf10 = []  
nb_impairs_sup10 = []
```

- vous pouvez ajouter des éléments à vos listes en utilisant la méthode `append()`
- vous pouvez utiliser l'opérateur modulo `%` qui renvoie le reste de la division entière entre deux nombres et dont voici quelques exemples d'utilisation :

```
Entrée [1]: 5 % 3
```

```
Out[1]: 2
```

Cette commande veut dire que si on divise 5 par 3, la différence entre le résultat de la division entière la plus grande (3) et le nombre initial (5) est de 2

Exercice 2 : Utilisation des boucles FOR et WHILE

Soit la liste de valeurs d'azimuts (exprimés en degrés) initiale

```
azm_in = [-130, 190, 370, 225, -200, -180]
```

Construisez une boucle qui permet d'afficher une liste de ces azimuts dans l'intervalle [-180 ;180]. Exprimez les azimuts calculés en radians.

Exercice d'application : La bataille navale simplifiée

Créer une boucle qui permette de trouver un bateau caché dans une grille de 5 par 5.

Consignes :

- Si le joueur donne un numéro de ligne et de colonne en dehors de la grille, alors il reçoit un message d'erreur
- Le joueur a 10 coups pour trouver le bateau, sinon il perd
- Lorsque le joueur tire dans l'eau on affiche « A l'eau ! », lorsqu'il touche le bateau, on affiche « Coulé ! »

Indication, pour afficher un bloc d'invite de commande on utilisera la commande

```
Entrée [ ]: var = int(input("Inviter l'utilisateur à saisir un nombre"))
```

