

# Exercices pour la séance de révision

Abiola Adeye: abiola.adeye@epfl.ch

2022/10/31

## Semaine 1

### Question 1: (🕒 15 minutes) Soustraction de nombres binaires

Effectuer les opérations suivantes :

1.  $01111111_{(2)} - 01000000_{(2)}$
2.  $10000000_{(2)} - 00000001_{(2)}$
3.  $10101010_{(2)} - 01010101_{(2)}$

## Semaine 2

**Question 2:** (🕒 10 minutes) En utilisant l'invite de commande (Terminal), exécutez le programme Python suivant en lui passant des paramètres.

```
1 import sys
2
3
4 if __name__ == '__main__':
5     arguments = sys.argv
6     if len(arguments) == 3:
7         print("La somme de {} et {} est {}".format(sys.argv[1], sys.argv[2], int(sys.argv[1])+int(sys.argv[2])))
8     else:
9         print("Assurez-vous de passer deux arguments à votre programme")
```

## Semaine 3

### Question 3: (🕒 10 minutes) Floating point

Voici la représentation en binaire d'un nombre à virgule flottante :

signe	exposant	mantisse
0	10110101	010000010000000000000001

Que vaut cette représentation en base 10? Utiliser la représentation des floating points (avec un biais de 127). Arrondir les résultats intermédiaires et la valeur finale au 3ème chiffre significatif après la virgule.

## Semaine 4

### Question 4: (🕒 5 minutes) Portée des variables (Python)

Qu'affiche le programme suivant ?

```

1 x = 2
2
3 def fonction():
4     x = 3
5
6     def fonction2():
7         global x
8         print("x=" + str(x))
9
10    fonction2()
11    print("x=" + str(x))
12
13 print(fonction())

```

### >\_ Solution

```

x=2
x=3
None

```

## Semaine 5

### Question 5: (🕒 10 minutes) Manipulation des listes en Java

Créez une liste nommée `ma_liste` contenant les nombres 1,2,3,4 et 5. Affichez le deuxième élément de la liste ainsi que la taille de la liste.

Créez une liste `ma_liste_m` liée à la liste `ma_liste`. Ajoutez le chiffre 6 à la fin de la liste, et le chiffre 0 au début de cette dernière.

Ajoutez ceci au début de votre code :

```

1
2 import java.util.List;
3 import java.util.LinkedList;

```

Ajoutez ceci à la fin de votre code :

```

1 for(int i=0;i<ma_liste_m.size();i++){
2     System.out.println(ma_liste_m.get(i));
3 }

```

## Semaine 6

### Question 6: (🕒 10 minutes) Création de classe et encapsulation

Créez une classe `Dog` contenant les attributs suivants :

1. Un attribut `public String` nommé `name`
2. Un attribut `private List` nommé `tricks`
3. Un attribut `private String` nommé `race`
4. Un attribut `private int` nommé `age`
5. Un attribut `private int` nommé `mood` initialisé à 5 (correspondant à l'humeur du chien)
6. Un attribut de classe (`static`) `private int` nommé `nb_chiens`

Créez une méthode publique du même nom que la classe (`Dog`). Cette méthode est appelée le `constructeur`, elle va servir à initialiser les différentes instances de notre classe. Un `constructeur` en `Java` aura le même nom que la classe, et le `constructeur` en `Python` sera défini par la méthode `__init__`. Cette méthode prendra en argument les éléments suivants qui seront utilisés pour initialiser les attributs de notre instance :

1. Une chaîne de caractères `name`,
2. Une liste `tricks`,

3. Une chaîne de caractères **race**,
4. Un entier **age**.

Pour finir, cette méthode doit incrémenter l'attribut de classe **nb\_chiens** qui va garder en mémoire le nombre d'instances créées.