**Swift, bases du langage**

**Exercice**

1. **Syntaxe de base**

**Créer un projet « base » avec un seul fichier. Ecrire une fonction par exercice et l’appeler dans le « main ».**

1. Créer une fonction « Tri » qui trie trois entiers donnés dans l’ordre décroissant.
2. Créer une fonction « Boucle » qui affiche une suite d’entiers pairs entre deux bornes incluses rentrées au clavier.
3. Créer une fonction « FizzBuzz » qui affiche la liste des 100 premiers entiers en remplaçant les multiples de 3 par fizz, les multiples de 5 par buzz et les multiples de 15 par fizzbuzz. Tous les 10 entiers, un saut de ligne est imposé.
4. Afficher le 4eme caractère de la chaîne de caractères rentrée au clavier si cela est possible sinon -1.
5. Afficher la mot inverse de la chaîne de caractères rentrée au clavier.
6. Créer un projet « Palindrome » qui indique si la chaîne rentrée au clavier un palindrome ou non.
7. Créer une fonction qui affiche un élément du tableau par ligne et qui prend en paramètre un tableau d’entier nommé tab. Tester la fonction dans le main.
8. Créer une fonction qui affiche le couple indice : valeur par ligne pour chaque ligne du tableau et qui prend en paramètre un tableau d’entier nommé tab.
9. Ecrire une fonction qui retourne le plus petit entier d’un tableau d’entier à une dimension.
10. Ecrire une fonction qui retourne l’indice du plus petit élément d’un tableau d’entier à une dimension.
11. Ecrire une fonction qui calcule la somme des éléments d’un tableau d’entier à une dimension.
12. Ecrire une fonction qui calcule la moyenne des éléments d’un tableau d’entier à une dimension.
13. Ecrire une fonction qui prend un tableau d’entier à une dimension nommé tab et un entier n et qui retourne si l’entier n est bien ou non dans le tableau.
14. Ecrire une fonction qui prend un tableau d’entier à une dimension nommé tab et un entier n et qui retourne l’indice de l’entier n ou -1 si celui n’est pas dans le tableau.
15. Ecrire une fonction qui prend un tableau d’entier à une dimension nommé tab et un entier n et qui retourne le nombre d’occurrences de l’entier.
16. Créer une fonction qui prend en paramètre un tableau tab d’entier à une dimension et qui retourne un tableau correspondant à l’inverse du tableau passé en paramètre.
17. Créer une fonction qui prend en paramètre un tableau tab d’entier à une dimension et qui retourne un tableau composé uniquement des éléments d’indice pair du tableau passé en paramètre.
18. **Objets**

Créer un projet « Zoo ». Un zoo est un ensemble d’animaux. Un animal est caractérisé par son nom, son âge, son genre (male, femelle) et son espèce. Un animal peut vieillir et mettre bas uniquement si c’est une femelle (renvoi un nouvel animal de la même espèce d’âge 1).

Créer une classe/structure Animal pour représenter les animaux. Créer une classe/structure Zoo qui va contenir une collection d’Animaux ainsi qu’un dictionnaire qui indique combien il y a d’animaux par espèce.

Dans votre programme, peupler votre zoo avec une vingtaine d’animaux diverses dont des femelles, faire naître une dizaine d’animaux et faire disparaître les quatre plus vieux animaux (ne pas oublier que cela va impacter le dictionnaire automatiquement).

**Animal = classe**

**No string, age int, genre enum, espece string**

**Zoo = classe**

**Attribut : animaux [] dico rencensement**