

## TP 3 - Boucles

---

### Exercice 1 : Somme de termes

---

**Q1.1** Écrire en VBA une macro qui demande à l'utilisateur de saisir deux entiers et qui affiche la somme des nombres compris entre ces deux nombres, les deux nombres inclus. Attention, le premier entier saisi par l'utilisateur peut être plus grand que le second.

**Q1.2** Modifier la macro précédente pour qu'elle ne fasse la somme que des termes pairs. Il est demandé de n'effectuer qu'un seul test de parité dans toute la macro.

---

### Exercice 2 : Sphinx

---

Écrire une macro VBA permettant de jouer au jeu suivant : un premier joueur choisit un nombre positif et l'indique à l'ordinateur à l'aide d'une boîte de dialogue (`Application.InputBox`). Un deuxième joueur essaie de deviner ce nombre en proposant des nombres (toujours avec `Application.InputBox`). A chaque tentative le programme indique si le nombre à trouver est plus petit ou plus grand que le nombre du deuxième joueur. La partie s'achève lorsque le deuxième joueur trouve la bonne solution. Le programme affiche alors :

*Bravo ! Le nombre est bien ... Vous l'avez trouvé en ... coups.*

---

### Exercice 3 : Masterspirit

---

Le but de cet exercice est d'écrire un programme qui simule le jeu entre deux personnes suivant :

- une première personne retient un nombre compris entre 100 et 999. Le programme demande à cette personne son nombre par un `Application.InputBox`.
- La deuxième personne essaie de deviner ce nombre. Elle précise un nombre avec `Application.InputBox`.

*Attention* : pour simplifier l'exercice, on considère que les deux joueurs ne donnent que des nombres avec des chiffres différents (aucun d'eux ne va dire 252 par exemple) et on fait l'hypothèse que les joueurs respectent nos contraintes sur les nombres, donc pas besoin de vérification sur la validité du nombre.

**Q3.1** Dans un premier temps, écrire la partie du programme qui demande aux deux joueurs leur nombre.

**Q3.2** Le programme va dire maintenant avec `MsgBox` au deuxième joueur combien de chiffres sont bien placés et combien de chiffres existent dans le premier nombre mais ne sont pas à la bonne place. Par exemple :

le nombre du premier joueur est 123

le nombre du deuxième joueur est 152 : « il y a 1 chiffre bien placé et 1 chiffre mal placé » (car 1 est au bon endroit et 2 n'est pas à la bonne position).

Écrire la suite du programme qui indique au deuxième joueur combien de chiffres sont bien placés et combien de chiffres existent dans le premier mais sont en mauvaise position. Pour l'instant le deuxième joueur ne fait qu'un seul essai.

**Q3.3** Compléter maintenant le programme pour qu'il demande au deuxième joueur de donner un nombre jusqu'à ce qu'il trouve le nombre du premier joueur. C'est à dire :

- si les 3 chiffres sont au bon endroit le programme s'arrête en affichant « bravo vous avez trouvé, le nombre est ... »
- sinon le programme demande un autre nombre au deuxième joueur jusqu'à ce qu'il trouve le bon nombre.

**Exemple :** le nombre du premier joueur est 123

Essai 1 du deuxième joueur : 456 : « il y a 0 chiffre bien placé et 0 chiffre mal placé »

Essai 2 du deuxième joueur : 912 : « il y a 0 chiffre bien placé et 2 chiffres mal placés »

Essai 3 du deuxième joueur : 152 : « il y a 1 chiffre bien placé et 1 chiffre mal placé »

Essai 4 du deuxième joueur : 129 : « il y a 2 chiffres bien placés et 0 chiffre mal placé »

Essai 5 du deuxième joueur : 123 : « il y a 3 chiffres bien placés et 0 chiffre mal placé. Bravo vous avez trouvé, le nombre est 123 ».