# Exercices chapitre 3 - Répétitives et expressions logiques

* Pour ce chapitre, à moins d’indications contraires, vous n’avez pas à faire la compréhension du problème et le jeu de tests.

## Exercice 3.1 - Le compte à rebours

Créer un programme qui effectue un compte à rebours à partir d’un nombre demandé à l’usager. Par exemple, un compte à rebours à partir de 10 affichera: 10 9 8 7 6 5 4 3 2 1 0.

## Exercice 3.2 - La somme d’une série

Calculer la somme d’une série de nombres d’un minimum à un maximum inclusivement. Par exemple, si on entre les valeurs 1 et 5, le programme doit retourner 15 (1+2+3+4+5). Testez votre programme avec 1 et 100. Quelle sera la réponse ?

## Exercice 3.3

Écrire un programme qui calcule la somme des nombres pairs de 2 à 32 inclusivement et la somme des nombres impairs de 33 à 65. Votre programme doit calculer la somme totale et l’afficher à la console. Essayez de réécrire ensuite votre programme en une seule boucle.

## Exercice 3.4

Écrire un programme qui calcule la somme des nombres d'un minium à un maximum inclusivement en incrémentant de 2 nombres à la fois. Par exemple, si on entre les valeurs 1 et 5, le programme doit afficher 1+3+5=9. Si on entre les valeurs 2 et 5, le programme doit afficher (2+4=6).

## Exercice 3.5

Écrivez un programme qui affiche une ligne de “o” en utilisant une boucle. Par exemple, si on entre le chiffre 10, on obtiendra à l’écran:

oooooooooo

Avec 5, on obtient:

ooooo

## Exercice 3.6 – Boucle dans une boucle

Écrire un programme qui demande un nombre entre 1 et 40 et qui affiche un motif (la lettre “o” par exemple). Voici un exemple de motif avec nbreLigne=3, nbreLigne=5 et nbreLigne=10.

|  |  |  |
| --- | --- | --- |
| nbreLigne=3 | nbreLigne=5 | nbreLigne=10 |
| oooooo | ooooooooooooooo | ooooooooooooooooooooooooooooooooooooooooooooooooooooooo |

## Exercice 3.7 - Les nombres premiers

Écrivez un programme qui demande à l’utilisateur un nombre entier entre 1 et 100 inclusivement (votre programme doit demander un nombre, tant et aussi longtemps que l'usager n'entre pas un nombre valide (entre 1 et 100).) et détermine si ce nombre est premier ou non. Un nombre premier est un nombre qui se divise uniquement par 1 et par lui-même. Par exemple, pour vérifier si le nombre 7 est premier, on divise par tous les nombres de 2 à 6 inclusivement et on vérifie si le reste de la division (Ça vous penser à quelque chose) est 0. Si le reste est 0, le nombre n’est pas premier.

7 / 2 = 3 reste 1, 7 / 3 = 2 reste 1, 7 / 4 = 1 reste 3, 7 / 5 = 1 reste 2, 7 / 6 = 1 reste 1

Comme aucun reste n’a été égal à 0, le nombre 7 est premier.

8 est-il premier ? 8 / 2 = 4 reste 0 ==> Comme le reste est 0, ce n’est pas un nombre premier.

## Exercice 3.8 – Boucle do - while

Un menu doit afficher 3 choix à l’usager:

1. Afficher “Bonjour”
2. Afficher “Hi”
3. Quitter le programme

Le programme doit attendre que l’usager entre un choix. Il doit recommencer la lecture du choix tant que le choix n’est pas valide. Le programme effectue le choix de l’usager et recommence tant que l’usager ne veut pas quitter le programme.

Réalisez cet exercice de deux façons, une fois avec des boucles while; et une fois avec des boucles do ... while.