

# EEIA 2021 | Informatique

## Exercices sur les tests et les boucles

*Attention : Les exercices ne sont pas classés par ordre de difficulté !!*

### I. Pour commencer...

1. Ecrivez un programme qui affiche “PAIR” si un nombre rentré par l'utilisateur est pair et “IMPAIR” sinon. Trouvez 2 façons différentes d'écrire un programme qui fait la même chose.
2. Ecrire un programme qui prend un entier positif  $n$  en entrée, et affiche tous les nombres de 1 à  $n$  qui sont divisibles par 3 mais pas divisibles par 5.
3. Faites un programme qui demande un nombre à l'utilisateur et indique si le nombre est premier ou non. Si le nombre  $n$  est inférieur à 10, le programme affichera  $n$  fois le texte “bonjour”.
4. Déclarer la liste `lst` suivante : `[14, 7, 6, 12, 2, 3, 3, 10]` et l'afficher.
5. Ensuite, modifier la liste de telle façon que son dernier élément soit divisé par 2 et la réafficher.
6. Modifier la liste de telle façon que l'on retranche 1 à tous ses éléments et la réafficher.
7. Afficher tous les éléments de la liste sur des lignes différentes.
8. Faire afficher seulement les nombres pairs (sans fabriquer de nouvelle).
9. Faire afficher 10 fois chaque élément de la liste sur la même ligne séparée par des espaces.
10. Afficher chaque élément autant de fois que sa valeur. Par exemple, 14 sera affiché 14 fois.

### II. Entraînement...

1. Ecrivez un programme qui demande l'âge de l'utilisateur et lui dit s'il est mineur ou majeur.
2. Ecrivez un programme qui calcule la somme des nombres allant de 1 à 1000.
3. Ecrivez un programme qui calcule la somme des nombres multiple de 7 allant de 1 à 1000.
4. A l'aide d'un programme, calculez le plus petit entier  $n$  tel que  $1 + 2 + \dots + n \geq 2021$ .

5. Écrivez un programme qui affiche la factorielle d'un nombre entier  $n$  prédéfini. On proposera deux solutions : une avec la boucle **for** et l'autre avec la boucle **while**.
6. Écrivez un programme qui, pour tous les entiers de 1 à 50, affiche **foo** si l'entier est multiple de 3, **bar** s'il est multiple de 5 et **foo bar** s'il est à la fois multiple de 3 et de 5.
7. Pouvez-vous écrire un programme qui calcule le nombre de façons différentes d'obtenir 20 en faisant la somme des valeurs des faces visibles hautes de 5 dés différenciés ? On considère des dés à 6 faces numérotées de 1 à 6.
8. Ecrivez un programme qui affiche la somme des chiffres d'un nombre entier  $n$  qu'on lui donne.
9. Ecrivez un programme qui laisse l'utilisateur rentrer autant de nombres positifs qu'il veut, tant que chaque nombre est strictement supérieur à celui d'avant. Si un nombre rentré ne vérifie pas la règle, le programme s'arrête et indique combien de nombres comporte la suite strictement croissante rentrée par l'utilisateur.
10. On rappelle qu'un entier  $n \geq 2$  est premier quand il n'est divisible que par 1 et lui-même : 7 est premier, mais  $6 = 2 \times 3$  ne l'est pas.
  - Faites un programme qui demande un nombre à l'utilisateur et indique si le nombre est premier ou non.
  - Faites un programme qui demande un nombre  $n$  à l'utilisateur et indique tous les nombres premiers qui sont inférieurs ou égaux à  $n$ .

### III. Toujours plus...

1. Ecrivez un programme qui demande deux nombres  $a$  et  $b$  à l'utilisateur et affiche son PGCD en utilisant l'algorithme d'Euclide.
  2. Ecrire un programme qui permet à l'utilisateur de saisir autant de nombres entiers qu'il veut et qui calcule la moyenne des nombres rentrés. Quand l'utilisateur a terminé, il écrit **stop** au lieu d'un nombre.
  3. Modifier le programme précédent pour qu'il redemande les nombres tant qu'ils ne sont pas compris entre 0 et 20.
  4. Que fait la fonction **mystere** ci-dessous quand on lui passe un entier  $n$  ?
- ```
def mystere (n):
    for i in range(2, n-1):
        if n%i == 0:
            return False
    return n>1
```
4. Chaque nouveau terme d'une suite de Fibonacci est généré en additionnant les deux termes précédents. En commençant avec 1 et 2, les 10 premiers de

la suite seront : 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, .... En considérant les termes de la suite de Fibonacci dont les valeurs n'excèdent pas  $4 \times 10^6$ , calculez la somme des termes de valeurs paires.

5. On représente l'équation du second degré  $ax^2 + bx + c = 0$  par le triplet  $(a, b, c)$ . Ecrivez un programme qui demande trois nombres  $a$ ,  $b$  et  $c$  à l'utilisateur et affiche la ou les solutions réelles de l'équation du second degré associé.
6. Faites la même chose que précédemment mais en affichant cette fois-ci les deux solutions complexes de l'équation.