

# Exercices

## Structures itératives



1. Ecrire un programme avec une boucle **for** qui compte de **0** à **30**, de **1** en **1**. Les nombres seront affichés les uns au-dessous des autres.
2. Ecrire un programme avec une boucle **for** qui compte de **0** à **30**, de **1** en **1**. Les nombres seront affichés les uns à côté des autres.
3. Ecrire un programme avec une boucle **for** qui compte de **0** à **30**, de **6** en **6**.
4. Ecrire un programme avec une boucle **for** qui compte de **70** à **200**, de **6** en **6**.
5. Ecrire un programme avec une boucle **for** qui compte à l'envers de **30** à **5**, de **1** en **1**.
6. Ecrire un programme avec une boucle **for** qui compte à l'envers de **30** à **0**, de **3** en **3**.
7. Utiliser la fonction **sleep** du module **time** pour réaliser un compte à rebours, toutes les secondes, à partir de **40**. Les nombres seront affichés les uns au-dessous des autres, puis les uns à côté des autres. A la fin du programme, le message **FIN DU COMPTE A REBOURS** s'affichera.
8. Tester le code suivant et expliquer en quoi il pose problème.

```
x = 0
while x < 1 :
    print("Je ne m'arrête jamais")
```
9. Écrire un programme avec une boucle **for** qui affiche les **20** premiers termes de la table de multiplication par **7**, en signalant au passage ceux qui sont des multiples de **3**.
10. Écrire un programme à l'aide d'une boucle **for** qui affiche une table de conversion de sommes d'argent exprimées en euros, en dollars canadiens, de **1** à **10** euros.
11. Même question avec une boucle **while**.
12. Modifier le programme pour qu'il s'arrête à la somme de avec **1213.53** dollars canadien.
13. Écrire un programme qui affiche une suite de **12** nombres dont chaque terme est égal au triple du terme précédent.
13. Ecrire un programme avec une boucle **while** infinie qui demande un prix HT et affiche sa valeur TTC.
15. Modifier le programme pour qu'il s'arrête si l'utilisateur entre **0** à la place du prix HT.
16. Modifier le programme pour qu'il s'arrête si l'utilisateur entre **FIN** à la place du prix HT.

17. Modifier le programme pour qu'il propose l'option **1. TVA à 5.5 %** ou l'option **2. TVA à 20 %**.

18. Ecrire un programme où l'utilisateur donne un entier positif et le programme annonce combien de fois de suite cet entier est divisible par **2**.

19. On dispose d'une feuille de papier d'épaisseur **0.1 mm**. Ecrire un programme qui détermine combien de fois on doit plier la feuille au minimum pour que son épaisseur dépasse **400 m**.

20. Avec une double boucle **for**, écrire un programme qui affiche la suite de couples de nombres ci-contre.

21. Modifier le programme pour que l'affichage soit celui ci-dessous.

```
( 0 , 0 ) ( 0 , 1 ) ( 0 , 2 ) ( 0 , 3 ) ( 0 , 4 )
( 1 , 0 ) ( 1 , 1 ) ( 1 , 2 ) ( 1 , 3 ) ( 1 , 4 )
( 2 , 0 ) ( 2 , 1 ) ( 2 , 2 ) ( 2 , 3 ) ( 2 , 4 )
```

22. Modifier le programme pour que l'affichage soit celui ci-dessous.

```
0 1 2 3 4 5 6 7 8 9
10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28 29
30 31 32 33 34 35 36 37 38 39
40 41 42 43 44 45 46 47 48 49
50 51 52 53 54 55 56 57 58 59
60 61 62 63 64 65 66 67 68 69
70 71 72 73 74 75 76 77 78 79
80 81 82 83 84 85 86 87 88 89
90 91 92 93 94 95 96 97 98 99
```

23. Ecrire un programme qui donne en colonnes toutes les tables de multiplication de **1 à 10**.

24. Écrire un programme avec une boucle **for** qui affiche les **100** premiers nombres premiers.

25. Écrire un programme avec une boucle **while** qui affiche les **100** premiers nombres premiers.

```
( 0 , 0 )
( 0 , 1 )
( 0 , 2 )
( 0 , 3 )
( 0 , 4 )
( 1 , 0 )
( 1 , 1 )
( 1 , 2 )
( 1 , 3 )
( 1 , 4 )
( 2 , 0 )
( 2 , 1 )
( 2 , 2 )
( 2 , 3 )
( 2 , 4 )
```