Exercices sur les boucles Tant que

Exercice 1

1. Écrire un programme qui :
   - choisit au hasard un entier A entre 1 et 10 ;
   - demande à l'utilisateur un entier X et affiche « gagné » si X=A, « perdu » sinon.

   Comme nous n'avons plus fait appel au module qui gère le hasard depuis un bon moment, on a écrit une petite partie du programme, que vous n'avez plus qu'à recopier et compléter dans l'éditeur de Python :

```
from random import *
A = randint(1,10)
.............
if X == A :
    .............
.............
    .............
```

2. Modifier le programme pour qu'il choisisse un nombre A entre 1 et 10 et affiche « trouvé » , « trop grand » ou « trop petit » selon le nombre X entré par l'utilisateur.
3. Modifier une nouvelle fois le programme de façon qu'il choisisse un nombre A entre 1 et 1000, affiche « trouvé » , « trop grand » ou « trop petit » ; puis redemande un nombre X à l'utilisateur jusqu'à ce qu'il trouve A.
4. Modifier une dernière fois le programme pour qu'il affiche en sortie le nombre de tentatives nécessaires pour trouver A.

Exercice 2

1. Le module time permet de gérer le temps. Essayer par exemple le code ci-dessous :

```
from time import *
a = time( )
sleep(3)
b = time( )
print(b - a)
```

   a était l'heure au début du code, puis il y a eu une pause de 3 secondes ; et enfin b était l'heure à la fin du code. La différence b−a à la fin vaut donc 3 secondes et des poussières.

2. Reprendre l'exercice 1 et modifier le code pour afficher en sortie le temps nécessaire pour trouver le bon nombre.

3. Éditer un programme qui demande à l'utilisateur successivement 30 multiplications choisies au hasard (entre 2×2 et 10×10) et lui laisse une minute pour répondre correctement à toutes les questions. En sortie le programme affiche « gagné » ou « perdu ».

   On aura notamment besoin d'un petit bout de code de la forme

```
x = randint(2,10)
y = randint(2,10)
z = int(input(str(x)+"*"+str(y)+"="))
```

   avec la conversion des nombres aléatoires x et y en chaînes de caractères [1].

1. Type *string*, abrégé en *str*.