

TP1 - Les types de base et if - En plus

M. Tellene

EXERCICE 1

Pour cet exercice, créer un fichier `exercice_5.py`

Écrire un programme qui indique l'état de l'eau en fonction d'une température, c'est-à-dire "SOLIDE", "LIQUIDE" ou "GAZEUX".

On rappelle que :

- si la température est strictement négative, l'eau est à l'état solide.
- si la température est entre 0 et 100 (compris), l'eau est à l'état liquide.
- si la température est strictement supérieure à 100, l'eau est à l'état gazeux.

EXERCICE 2

Pour cet exercice, créer le fichier `exercice_6.py`

Écrire un programme qui en fonction d'une année donnée, indique s'il s'agit d'une année bissextile. On rappelle qu'une année est bissextile si elle est multiple de 4 mais pas multiple de 100, ou si elle est multiple de 400.

EXERCICE 3

Pour cet exercice, créer un fichier `exercice_7.py`

En géométrie, l'inégalité triangulaire est le fait que, dans un triangle, la longueur d'un côté est inférieure ou égale à la somme des longueurs des deux autres côtés. En somme si nous avons trois valeurs : AB, AC et BC et que nous avons les trois conditions suivantes réunies alors ABC forment un triangle :

- $AB \leq AC + BC$
- $AC \leq AB + BC$
- $BC \leq AB + AC$

Écrire un programme qui, avec 3 valeurs (AB, AC et BC), affiche si le triangle ABC peut exister. Pour tester que les trois conditions sont vraies, vous pouvez faire des instructions `if` imbriquées (des `if` dans des `if`) ou utiliser le mot-clé `and`.

Pour plus d'informations sur le `and`, vous pouvez consulter les sites suivants : <https://cscircles.cemc.uwaterloo.ca/9-fr/> ou <https://realpython.com/python-operators-expressions/#logical-operators>

EXERCICE 4

Pour cet exercice, il faudra modifier le fichier `exercice_7.py`

Vous avez écrit un programme qui détecte si 3 valeurs peuvent former un triangle. Rendons ce programme encore plus précis!

Pour ce faire, en plus d'indiquer si le triangle peut exister, le programme devra afficher la nature du triangle : équilatéral, isocèle, rectangle ou quelconque.

EXERCICE 5

Pour cet exercice, il faudra modifier le fichier `exercice_7.py`

Si vous testez le programme précédent en mettant toutes les valeurs à 0, un petit problème apparaît. Cela vous paraît-il possible? De manière plus générale, il est difficile de faire un triangle avec un côté égal à 0... L'objectif de cet exercice est de corriger ce problème.

EXERCICE 6

Pour cet exercice, créer le fichier `exercice_8.py`

On considère un billard dont les dimensions sont données par les variables `longueur` et `largeur`, avec une seule bille de coordonnées (x, y) . On suppose que le coin inférieur gauche du billard a pour coordonnées $(0, 0)$ et le coin supérieur droit les coordonnées $(longueur, largeur)$.

Après application d'un vecteur de déplacement (d_x, d_y) les nouvelles coordonnées de la bille sont $(x + d_x, y + d_y)$, si ce déplacement n'implique pas de transpercer une paroi du billard. Sinon la bille effectue un rebond parfait sur chacune des parois rencontrées : la nouvelle direction est symétrique de l'ancienne par rapport à la paroi sur laquelle la bille rebondit, et la distance restant à parcourir est inchangée.

Écrire un programme qui en fonction des coordonnées de la bille et d'un vecteur de déplacement (d_x, d_y) , pour lequel on supposera $-longueur \leq d_x \leq longueur$ et $-largeur \leq d_y \leq largeur$, affiche les coordonnées de la bille à la fin du mouvement.