

Variables Booléennes

Exercice 1 : Evaluation

Evaluer le résultat de ces expressions :

- (True or False) and False
- True or (False and False)
- True or False and False
- (True or False) and False or (True and False)

Exercice 2 : Evaluation paresseuse

En utilisant python, évaluer les expressions suivantes (le résultat peut être True, False ou Error) :

- 0==0/0
- True and 0==0/0
- True or 0==0/0
- 0==0/0 or True

Une seule des évaluations est True. Laquelle et pourquoi ?

Exercice 3 : Simplification

- En utilisant les propriétés de l'algèbre de Boole, justifier les simplifications suivantes :
 - $a \text{ and } (a \text{ or } b) \Leftrightarrow a$
 - $a \text{ and not } b \text{ or } a \text{ and } b \text{ or not } a \text{ and } b \text{ or not } a \text{ and not } b \Leftrightarrow \text{True}$

Exercice 4 : Problème 1

On cherche à exprimer en langage python la phrase suivante :

« Je traverse la route s'il n'y a une voiture ni à droite ni à gauche »

On utilise les variables :

g: présence d'une voiture à gauche

d : présence d'une voiture à droite

T : traverser

Ecrire l'expression de T en fonction de g et d de deux manières différentes

Exercice 5 : Problème 2

Les sujets du concours arrivent au lycée dans un coffre à serrure électrique **S**, actionnée par un moteur, commandée par trois clés différentes (**a**, **b**, **c**), ces clés sont confiées à trois responsables : le directeur du centre d'examen, un responsable du rectorat, et le proviseur du lycée.

Pour ouvrir le coffre il faut au moins deux parmi les trois responsables soient présents.

1. Représenter le problème sous la forme d'une table de vérité.
2. Donner l'expression de S en fonction de a, b et c.

Structure conditionnelle : if

Exercice 6 : Trouver le maximum de 3 nombres

Spécifier et écrire un programme qui demande à un utilisateur de saisir 3 nombres distincts, puis qui affiche le plus grand d'entre eux. (Conseil : utilisez une variable nommée `maxi` pour stocker et afficher le maximum en fin d'algorithme).

Boucles

Exercice 7 : Programme mystère

À l'aide d'une feuille de brouillon, déterminer le résultat affiché par le programme suivant :

```
nbPersonnes = 3
nbJours = 4
prixSéjour = 0
prixJournée = 1
for loop in range(nbJours):
    prixSéjour = prixSéjour + prixJournée
    prixJournée = prixJournée + 2
prixSéjour = prixSéjour + 14
print("Prix par personne :", prixSéjour // nbPersonnes)
```

Exercice 8 : Boucles à connaître : adaptation

Dans cet exercice, il faut adapter les boucles à connaître afin d'obtenir un résultat désiré.

1. Ecrire un programme qui détermine la somme des nombres entiers impairs compris entre 1 et 101, bornes incluses.
2. Ecrire un programme qui détermine le produit des nombres entiers impairs compris entre 1 et 101, bornes incluses.
3. Ecrire un programme qui affiche le plus petit entier n tel que la somme des entiers de 0 à n dépasse 1001 ?
4. Ecrire un programme qui affiche combien de fois faut-il multiplier 2 par lui-même pour atteindre 1 000 000 000 ? (sans utiliser le logarithme)
5. Suite récurrente d'ordre 1 : On considère la suite u_n définie pour tout entier naturel n par :

$$u_0 = 1$$

$$u_{n+1} = u_n + 2n + 3$$

- a. Ecrire un programme qui affiche les dix premiers termes de la suite ;
- b. Vérifier par conjonction que pour $0 \leq n \leq 100 : u_n = (n + 1)^2$.

Exercice 9 : Simulation d'épidémie

Ecrire un programme doit d'abord lire un entier fourni par l'utilisateur, la population totale de la ville. Sachant qu'une personne était malade au jour 1 et que chaque malade contamine deux nouvelles personnes le jour suivant (et chacun des jours qui suivent), afficher à partir de quel jour toute la population de la ville sera malade.