

Booléens

Fiche de synthèse

① BOOLÉENS ET TABLES DE VÉRITÉ

Définition 1 (Booléen et table de vérité)

- Un **booléen** ne prend que deux valeurs : vrai, noté 1, ou faux, noté 0. En Python, `True` et `False`. Le nom vient de George Boole (1815-1864).
- Une **fonction logique** prend des booléens en entrée et renvoie un booléen.
- Sa **table de vérité** donne la sortie pour chaque combinaison des entrées : 2^n lignes pour n entrées.

Les comparaisons, `==`, `!=`, `<`, `>`, `<=` et `>=`, produisent des booléens.

② LES OPÉRATEURS

a	b	$\bar{a}$	$a \cdot b$	$a + b$	$a \oplus b$	$\overline{a \cdot b}$
0	0	1	0	0	0	1
0	1	1	0	1	1	1
1	0	0	0	1	1	1
1	1	0	1	1	0	0

Opérateur	Notation	Python	Vrai si
NON	$\bar{a}$	<code>not a</code>	a est faux
ET	$a \cdot b$	<code>a and b</code>	les deux sont vrais
OU	$a + b$	<code>a or b</code>	au moins un est vrai
OU exclusif	$a \oplus b$	<code>a ^ b</code>	exactement un est vrai
NON-ET	$\overline{a \cdot b}$	<code>not (a and b)</code>	les deux ne sont pas vrais

Le OU est **inclusif** : il vaut 1 quand les deux entrées valent 1, comme le « ou » de « on entre avec un billet ou une carte d'abonnement ». Le OU exclusif est le « ou » de « la mer ou la montagne », où l'on ne choisit pas les deux. Le ET donne le même résultat qu'une multiplication, d'où le point.

③ PROPRIÉTÉS

Propriété	ET	OU
élément neutre	$a \cdot 1 = a$	$a + 0 = a$
élément absorbant	$a \cdot 0 = 0$	$a + 1 = 1$
idempotence	$a \cdot a = a$	$a + a = a$
complément	$a \cdot \bar{a} = 0$	$a + \bar{a} = 1$
commutativité	$a \cdot b = b \cdot a$	$a + b = b + a$

- Double négation : $\bar{\bar{a}} = a$. Les deux opérateurs sont aussi associatifs.
- Distributivité : $a \cdot (b + c) = a \cdot b + a \cdot c$, et, sans équivalent en arithmétique, $a + b \cdot c = (a + b) \cdot (a + c)$.
- Absorption : $a + a \cdot b = a$ et $a \cdot (a + b) = a$.

Théorème 1 (Lois de De Morgan)

$$\overline{a + b} = \bar{a} \cdot \bar{b} \quad \overline{a \cdot b} = \bar{a} + \bar{b}$$

« Ne pas (A ou B) » revient à « ne pas A et ne pas B ». En Python, `not (age < 18 or solde < 0)` s'écrit plus lisiblement `age >= 18 and solde >= 0`.

④ DES INTERRUPTEURS AUX TRANSISTORS

- Deux interrupteurs en **série** réalisent le ET, deux interrupteurs en **parallèle** le OU, un interrupteur normalement fermé le NON.
- Shannon a montré que l'algèbre de Boole s'applique aux circuits électriques. Dans un processeur, les **portes logiques** sont faites de **transistors**, des interrupteurs commandés électriquement.
- La porte NON-ET, ou NAND, est **universelle** : elle suffit à construire les trois opérateurs. Si l'on note $a \uparrow b = \overline{a \cdot b}$, alors $\bar{a} = a \uparrow a$, $a \cdot b = (a \uparrow b) \uparrow (a \uparrow b)$ et $a + b = (a \uparrow a) \uparrow (b \uparrow b)$.

À retenir.

- ET : les deux vrais ; OU : au moins un vrai ; OU exclusif : exactement un vrai.
- Une table de vérité à n entrées compte 2^n lignes.
- De Morgan : $\overline{a + b} = \bar{a} \cdot \bar{b}$ et $\overline{a \cdot b} = \bar{a} + \bar{b}$.
- Les portes logiques sont faites de transistors ; la porte NON-ET est universelle.