

Il est très fréquent lorsqu'on écrit un algorithme ou un programme qu'on doit adapter les instructions en fonction des événements qui se produisent. Il faut donc réaliser des tests en utilisant une instruction conditionnelle.

1. Instruction conditionnelle

■ Permet de tester une condition puis d'effectuer une (ou plusieurs) instruction(s) en fonction du résultat de ce test. Elle s'utilise sous la forme d'un bloc de la forme :

→ **si** **Condition** **alors** **Instruction(s)**.

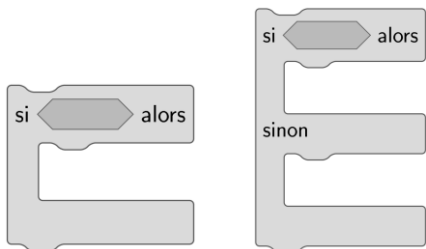
Dans ce cas, si la **Condition** est réalisée alors les **Instructions**

Exemple.

- Un conducteur se déplace en ville au volant de son véhicule. Lorsqu'il arrive à une intersection avec un feu tricolore, il s'arrête lorsque le feu est rouge.
 - Instruction conditionnelle : **si le feu est rouge alors le conducteur s'arrête.**

SCRATCH

■ 2. Pour **utiliser une instruction conditionnelle** :
Catégorie **Contrôle** et les blocs de commande utilisés sont :



👉 Pour tester la condition étudiée, on utilise des :

■ Opérateurs de comparaison (<, =, >)



■ Opérateurs logiques (et, ou, non)



SCRATCH

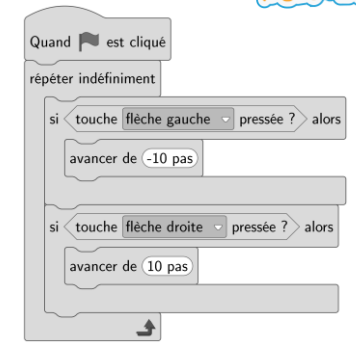
ChAP3. Utiliser les instructions conditionnelles

(AFC11)

3. Déplacement

Exemple.

- Écrire un programme dans Scratch qui permette de faire reculer un lutin de 10 si ← est pressée et de faire avancer de 10 si → est pressée.



4. Age

Exemple.

- Écrire un programme Scratch qui renvoie « mineur » ou « majeur » suivant l'âge qu'on lui renseigne.



5. Modulo

Exemple.

- Dans le programme ci-contre, on utilise la commande modulo qui donne le reste de la division euclidienne.

Autrement dit, $\text{réponse} \bmod 3 = 0$ est vrai, si réponse est divisible par 3, et faux sinon.

Qu'affiche le programme si :

1. on entre le nombre 27 ? Toc
2. on entre le nombre 41 ? TikTok
3. on entre le nombre 60 ? Tic
4. on entre le nombre 55 ? Tic Tac

