

Un **algorithme** est une **suite d'instructions ordonnées** qui permet de **résoudre un problème**.
 Un programme c'est la **traduction d'un algorithme** dans un **langage** que l'**ordinateur** comprend.



Variable

C'est un espace de **mémoire** pour **stocker** une **valeur** qui peut changer au cours du programme. Elle peut être de trois types : **Texte**, **Nombre** ou **Booléen** (Vrai ou faux, True/False en anglais). Une **instruction d'affectation** associe une valeur à une variable.

Instruction itérative

Permet de **répéter** un bloc d'instructions un certain nombre de fois ou indéfiniment. Exemples :
Tant que (while) : répète tant qu'une condition est vraie.
Pour (for) : répète pour un nombre déterminé de fois.

Instruction conditionnelle

Permet de tester **si** une ou plusieurs conditions sont **vraies** ou **fausses**, et de faire des choix en conséquence (**si**, **sinon**) (If, Else en anglais).

Opérateurs

Opérateurs logiques : utilisés pour combiner des expressions logiques (**ET**, **OU**) (AND, OR en anglais).
Opérateurs arithmétiques : permettent de faire des opérations mathématiques (addition, soustraction, etc.).
Opérateurs de comparaison : utilisés pour comparer deux valeurs (=, >, <, etc.).

Exemple d'algorithme : Une porte automatique.
 En présence d'une personne, la porte s'ouvre et la lumière s'allume.
 Après 5 secondes, la porte se ferme et la lumière s'éteint.

Début

Variable luminosité = Lire_Capteur_Luminosite()
Variable présence = Lire_Capteur_Presence()

Répéter indéfiniment

Si

présence = vrai **ET** luminosité < 10

Alors

ouvrir porte
 allumer lampe
 attendre 5 secondes
 fermer porte
 éteindre lampe

Fin Si

Fin Répéter

Fin

Listes

Une liste c'est une **structure de données** qui permet de **stocker** plusieurs valeurs de **même type** (liste de noms, liste de notes, etc.).
 Exemple : [voiture, vélo, avion, bateau, train]

Blocs d'instructions

C'est des **groupes d'instructions** qui sont **exécutés ensemble**, souvent associés à des structures conditionnelles ou itératives.

Évènement

Un évènement est une **action** qui se **produit pendant l'exécution du programme**, comme un clic de souris ou une pression sur une touche.

Déclenchement d'une séquence par un évènement

Quand un **évènement se produit**, une partie du programme (une **séquence d'instructions**) peut être **exécutée**. Exemple : dans un jeu, déclencher le saut d'un personnage lorsqu'on appuie sur la touche « espace ».

Variable

C'est un espace de **mémoire** pour **stocker** une **valeur** qui peut changer au cours du programme. Elle peut être de trois types : **Texte**, **Nombre** ou **Booléen** (Vrai ou faux, True/False en anglais). Une **instruction d'affectation** associe une valeur à une variable.

Instruction itérative

Permet de **répéter** un bloc d'instructions un certain nombre de fois ou indéfiniment. Exemples :
Tant que (while) : répète tant qu'une condition est vraie.
Pour (for) : répète pour un nombre déterminé de fois.

Instruction conditionnelle

Permet de tester **si** une ou plusieurs conditions sont **vraies** ou **fausses**, et de faire des choix en conséquence (**si**, **sinon**) (If, Else en anglais).

Opérateurs

Opérateurs logiques : utilisés pour combiner des expressions logiques (**ET**, **OU**) (AND, OR en anglais).
Opérateurs arithmétiques : permettent de faire des opérations mathématiques (addition, soustraction, etc.).
Opérateurs de comparaison : utilisés pour comparer deux valeurs (=, >, <, etc.).

Même exemple en programmation par blocs

lorsque vous cliquez sur

luminosité à valeur mesurée par le capteur de lumière

présence à valeur mesurée par le capteur de présence

pour toujours

si présence = vrai et luminosité < 10 alors

ouvrir porte

allumer lampe

attendre 5 sec

fermer porte

éteindre lampe

Listes

Une liste c'est une **structure de données** qui permet de **stocker** plusieurs valeurs de **même type** (liste de noms, liste de notes, etc.).
 Exemple : [voiture, vélo, avion, bateau, train]

Blocs d'instructions

C'est des **groupes d'instructions** qui sont **exécutés ensemble**, souvent associés à des structures conditionnelles ou itératives.

Évènement

Un évènement est une **action** qui se **produit pendant l'exécution du programme**, comme un clic de souris ou une pression sur une touche.

Déclenchement d'une séquence par un évènement

Quand un **évènement se produit**, une partie du programme (une **séquence d'instructions**) peut être **exécutée**. Exemple : dans un jeu, déclencher le saut d'un personnage lorsqu'on appuie sur la touche « espace ».