

# DomoScratch

**COMMENT CONTRÔLER UN  
HABITAT À DISTANCE, GÉRER  
L'ÉNERGIE ET ASSURER LA  
SÉCURITÉ ?**

**SÉQUENCE 14 - RESSOURCES**

**ACTIVITÉ 3**

**LES ÉTAPES POUR RÉALISER UN  
PROGRAMME**



## 1. DOMOSCRATCH : LES VARIABLES

Les variables représentent des grandeurs physiques (vent, température, éclairement naturel), ou des événements (pluie, locaux occupés, et détection de mouvement, de bris de glace ou de fumée).

Les variables passent d'une valeur à l'autre quand l'utilisateur clique sur l'image correspondante, en phase de test de programme par exemple.

La variable de l'éclairement change automatiquement avec l'horaire (voir dans le tableau ci-dessous). Les variables « Heure » et « Jour » sont modifiables.

La valeur initiale de la variable est la valeur de la variable quand le drapeau vert est cliqué

Image	Nom de la variable	Valeur initiale	Valeurs possibles de la variable
	Vitesse du vent	10 (10 km/h)	10, 20, 45 ou 70 (km/h)
	Température	20 (20°C)	-10, 0, 10, 20 ou 30 (°C)
	Eclairement	2 (Nuit claire)	1 (nuit noire, de 0 h à 3 H du matin), 2 (Nuit claire de 3 à 6 H et de 21 H à minuit) 3 (aube ou crépuscule, de 6 à 9 H et de 18 à 21 H ), 4 (ciel clair mais nuageux, de 9 à 12 H et de 15 à 18 H), 5 (plein soleil, de 12 à 15 h).
	Pluie	NON (pas de pluie)	OUI ou NON
	Locaux_utilisés	NON (personne dans les locaux)	OUI ou NON
	Mouvement_détecté	NON (aucun mouvement détecté)	OUI ou NON
	Bris_de_glace_détecté	NON (aucun bris de glace détecté)	OUI ou NON
	Fumée_détectée	NON (aucune fumée détectée)	OUI ou NON
	Heure	5 (5 h du matin)	0, 1, 2, 3, ...23 (H)
	Jour	1 (Lundi)	1, 2, 3, 4, 5, 6, 7 (pour lundi, mardi, ... dimanche)

**Remarque** : vous pouvez afficher les valeurs des variables en cochant la case correspondante :  Jour

Sa valeur apparaît alors sur la scène comme ceci : Jour 1

## 2. DOMOSCRATCH : LES ORDRES (CATÉGORIE « ÉVÉNEMENTS »)

L'état initial de l'actionneur est son état quand le drapeau vert est cliqué. 🚩



Actionneur	Messages (Ordres)	Image	État initial	Commentaires
Lumière intérieure	lumière_intérieure_off lumière_intérieure_on		Éteinte	
Lumière extérieure	lumière_extérieure_off lumière_extérieure_on		Éteinte	Les lumières extérieures servent à guider les personnes à l'extérieur du bâtiment
Arrosage automatique	arrosage_off arrosage_on		Inactif (pas d'arrosage)	
Volets roulants	volets_fermer_complètement volets_fermer_légèrement volets_ouvrir_complètement volets_ouvrir_légèrement		Fermés complètement	Ouvrir ou fermer les volets légèrement revient à les fermer ou les ouvrir d'un dixième
Auvent déployable	auvent_rentre auvent_sortir		Sortis entièrement	Un auvent déployable protège des rayons du soleil
Chauffage central	chauffage_off chauffage_on		Éteint (pas de chauffage)	Le chauffage central, quand il est en activité, permet de maintenir une température minimale de 20°C
Climatisation	climatiseur_off climatiseur_on		Éteint (pas de climatisation)	La climatisation, quand elle est en activité, permet de maintenir une température maximale de 20°C
Signal d'alarme	alarme_off alarme_on		Éteint (pas d'alarme)	L'alarme prévient les utilisateurs du bâtiment d'un danger. Plusieurs séquences d'alarme sont possibles.
Portes coupe-feu	porte_coupe_feu_off porte_coupe_feu_on		OFF : ouverte	En cas d'incendie, les portes coupe-feu se ferment pour éviter la propagation des flammes et fumées
Appel des pompiers	appeler_le_18		Pas d'appel pompier	
Appel de la police	appeler_le_17		Pas d'appel police	

### 3. Programmer ou coder c'est imaginer une solution automatique qui va résoudre un problème.

Le programmeur avant de réaliser un programme passe d'abord par l'étape création d'un algorithme. Cela lui permet d'avoir une idée claire de la logique de fonctionnement du programme qu'il va réaliser

#### 1. Décrire le fonctionnement à programmer en langage naturel

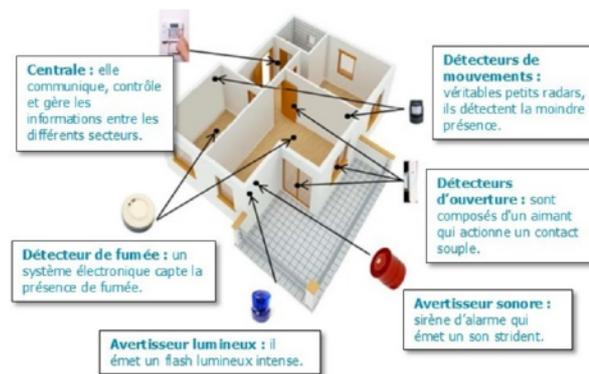
##### Algorithme en langage naturel du fonctionnement d'une alarme anti-intrusion

*Si quelqu'un franchit la porte ou une fenêtre de la maison, et si l'alarme est active alors une alarme sonore se déclenche.*

*L'alarme s'arrête si l'utilisateur désactive le système d'alarme.*

Mots clés en gras

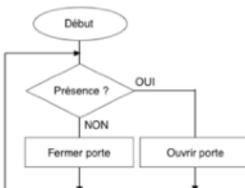
Le fonctionnement que vous programmez devra être logique. Dans le langage naturel, il faudra utiliser les éléments qui permettent de décrire cette logique, il faudra rechercher les mots tels que Si, ou, alors, et...



#### 2. Décrire l'algorithme de manière graphique avec un organigramme

##### Elaboration d'un algorithme

Si Présence  
Alors ouvrir porte  
Sinon fermer porte

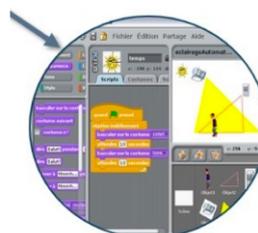


Si le problème à résoudre est simple on peut passer directement du langage naturel à la programmation.

Par contre si le problème est complexe il sera nécessaire de réaliser une version graphique de l'algorithme sous forme de logigramme.

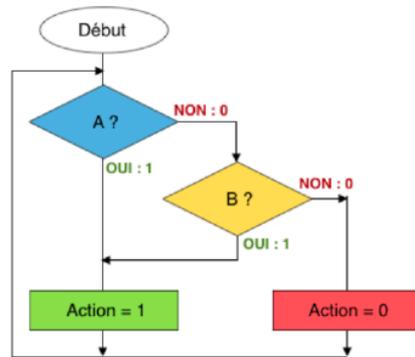
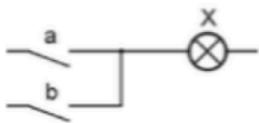
#### 3. Programmer de manière graphique ou avec des codes

Elaboration du programme à partir de l'algorithme



### 4. Représenter la fonction « ET » et la fonction « OU » dans un logigramme

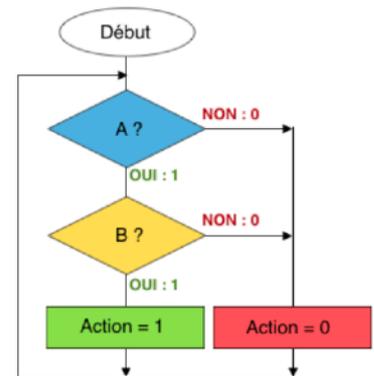
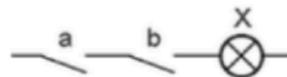
A ?	B ?	Sortie
0	0	0
0	1	1
1	0	1
1	1	1



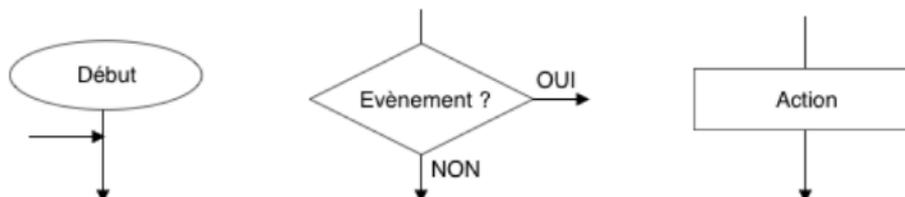
La fonction « ET »

La fonction « OU »

A ?	B ?	Sortie
0	0	0
0	1	0
1	0	0
1	1	1



### 5. Rappels sur la représentation des logigrammes



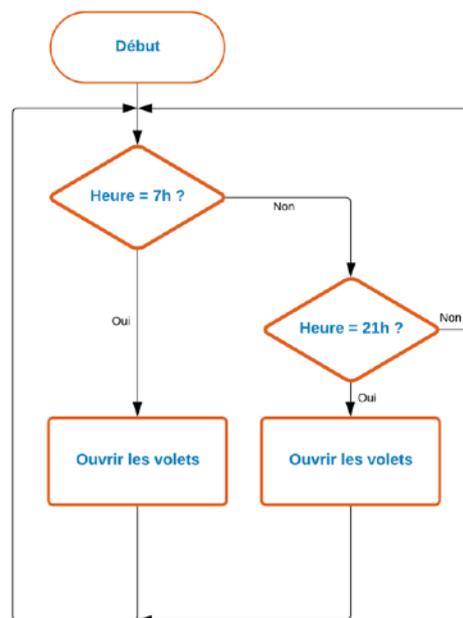
Un algorithme est une suite d'instructions précise et structurée qui décrit la manière dont on résout un problème. Cette description peut être textuelle (si, alors, sinon, tant que...) ou graphique (appelé également organigramme ou logigramme). Dans ce cas des normes d'écritures sont à respecter :

### 3. Exemple avec le programme A

#### 1. Le demande programmation

Le programme A doit : ouvrir les volets tous les jours le matin à 7h et les refermer le soir à 21h.

#### 2. Le logigramme du programme A



#### 3. Le programme sous Scratch 3 du programme A

