

# Notion de boucle et de condition avec

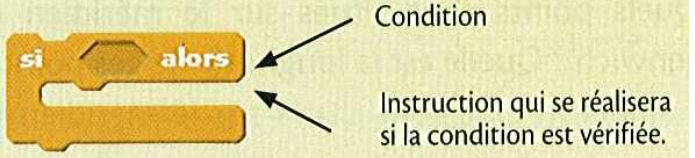
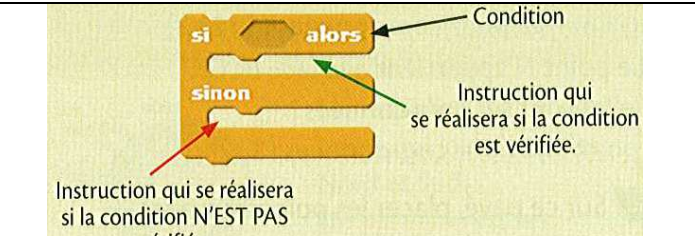
**Objectif :** Utiliser des structures itératives et conditionnelles.

## A connaître

### Instruction conditionnelle


Une **instruction conditionnelle** est une instruction qui a besoin d'une **condition** pour se réaliser. On reconnaît une instruction conditionnelle grâce aux mots **Si** et **Alors**. Dans SCRATCH, il y a deux blocs d'instruction conditionnelle :

**Si ... Alors ...** et **Si ... Alors ... Sinon ...**

<p><b>Si</b> ... condition ... <b>Alors</b> ... instruction ... <b>FinSi</b></p>	 <p>Condition</p> <p>Instruction qui se réalisera si la condition est vérifiée.</p>
<p><b>Si</b> ... condition ... <b>Alors</b> ... instruction A ... <b>Sinon</b> ... instruction B ... <b>FinSi</b></p>	 <p>Condition</p> <p>Instruction qui se réalisera si la condition est vérifiée.</p> <p>Instruction qui se réalisera si la condition N'EST PAS vérifiée.</p>

### Boucle

Quand on écrit un algorithme, il est parfois utile de répéter un certain nombre de fois les mêmes instructions. On utilise pour cela une structure de **boucle**. Dans une structure de boucle, toutes les instructions sont exécutées à chaque passage de la boucle.

<p>Les instructions dans la boucle ne sont effectuées que si la condition est vérifiée</p> <p><b>Tant que</b> ... condition ... <b>Faire</b> ... instruction ... <b>FinTantQue</b></p>	
<p>Cette boucle utilise un compteur et permet de répéter une série d'instructions, un nombre connu de fois</p> <p><b>Pour i allant de</b> (nb entier) <b>à</b> (nb entier) <b>Faire</b> ... instruction ... <b>FinPour</b></p>	

Dans SCRATCH, les commandes permettant d'exécuter des boucles et les conditions sont dans la catégorie **Contrôle**.

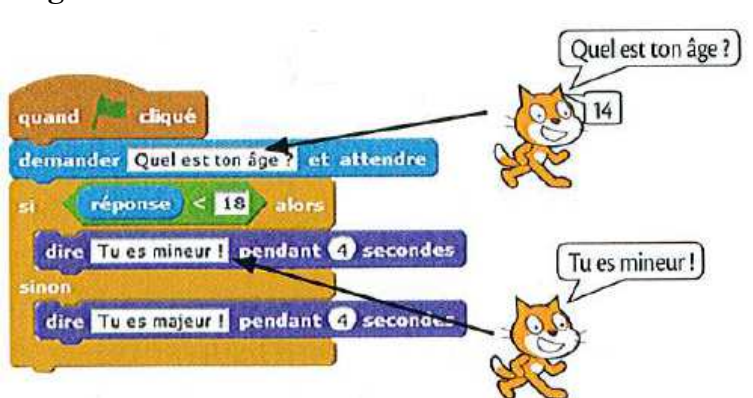
## Exercice 1 – Exemples d’instruction conditionnelle

Que font les programmes suivants ?

Programme 1



Programme 2



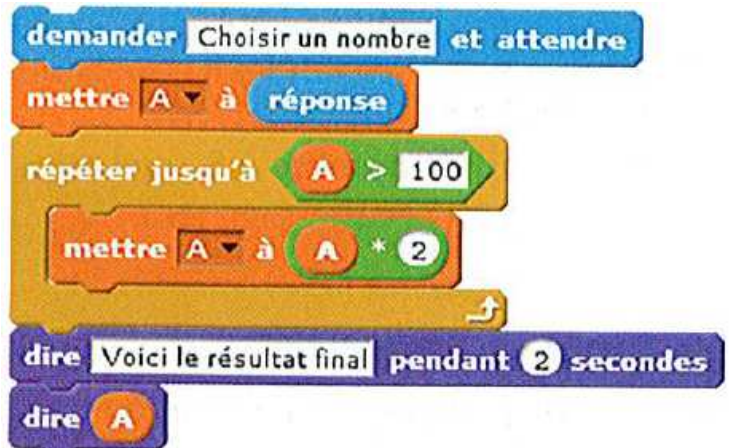
## Exercice 2 – Exemples de boucle

Que font les programmes suivants ?

Programme 3

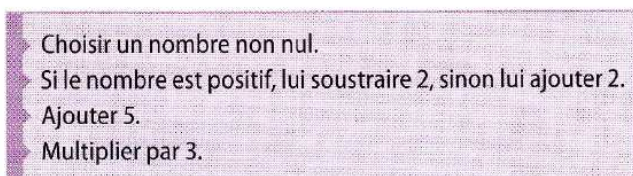


Programme 4

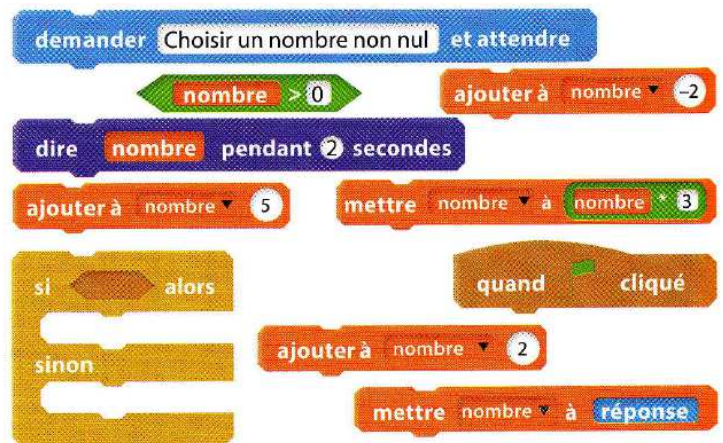


## Exercice 3 – Un programme de calcul

Voici un programme de calcul :



- Quel résultat ce programme donne-t-il si on choisit 3 au départ ?
- Quel résultat donne-t-il si on choisit -2 ?
- Écrire un script donnant le résultat de ce programme de calcul en utilisant les instructions ci-contre.



## Exercice 4 – Deux scripts simultanément

Avec SCRATCH, on peut définir plusieurs personnages (des « lutins »), chacun ayant son propre script, qui décrit son comportement.

Définir deux lutins, un gourmand comme cet insecte, et un appétissant comme la pomme, et leur associer les scripts ci-dessous.

### Script de l'insecte

### Script de la pomme

Que se passe-t-il quand vous cliquez sur le drapeau vert ?

## Exercice 5 – Corriger un algorithme

Une société de location de VTT propose le contrat suivant : un forfait de 12€ pour une durée d'utilisation inférieure à 2 heures et un ajout de 2,50€ par heure supplémentaire.

Minh propose un algorithme pour déterminer le coût  $C$  d'une location en fonction de la durée  $t$  de la location.

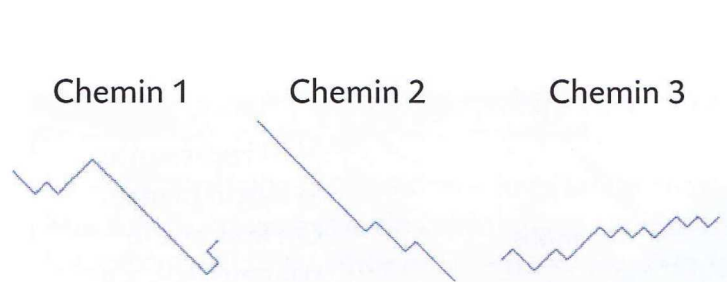
**Variables  $C, t$  : nombres**

Saisir  $t$   
**Si**  $t < 2$   
     **Alors**  $C$  reçoit  $12 + 2,5 \times t$   
     **Sinon**  $C$  reçoit 12  
**FinSi**  
 Afficher  $C$

- Sans utiliser l'algorithme, calculer les coûts pour une durée de 1 heure, puis une durée de 4 heures.
- En utilisant l'algorithme, refaire les mêmes calculs.
- Cet algorithme est-il correct ? Sinon, le corriger.
- Ecrire le programme SCRATCH correspondant.

## Exercice 6 – Une marche aléatoire **BREVET**

Avec le script SCRATCH proposé, le lutin va se déplacer de manière aléatoire.



Parmi les trois dessins proposés, un n'est certainement pas possible avec ce script. Lequel ? Parmi les deux autres scripts, lequel a été produit par le script ? Explique ton raisonnement.