

# Les premiers pas en programmation avec Scratch

## 1. Quelques notions à connaître

Le but de la programmation est de rendre automatique des opérations répétitives ou de résoudre un problème qui nous est posé. Dans tous les cas, il faut écrire un algorithme.

Un **algorithme** est une suite d'**instructions** à appliquer dans un ordre logique pour résoudre un problème et obtenir un résultat. Il peut se représenter de multiples façons,

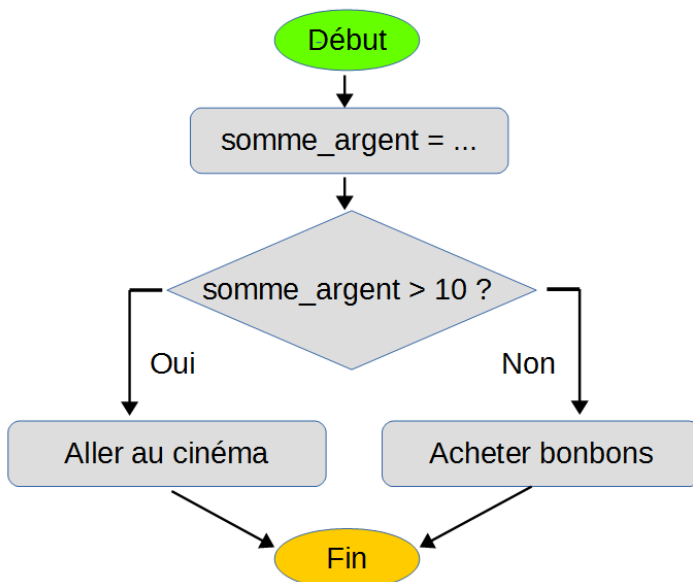
Voici trois représentations sur un exemple très simple :

*J'ai une somme d'argent en poche. Si j'ai plus de 10€ je vais au cinéma, sinon je m'achète un paquet de bonbons.*

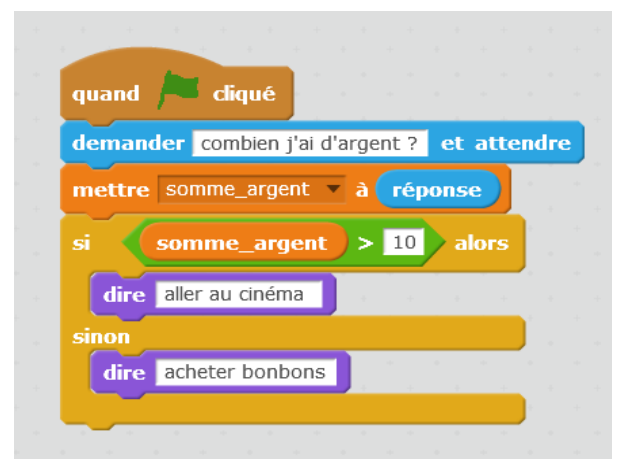
Représentation en « **pseudo code** » (du quasi-français)

```
Début
Compter somme_argent
Si somme_argent > 10
    Alors aller au cinéma
Sinon acheter bonbons
Fin
```

Avec un **organigramme**



Avec Scratch



Dans les 3 représentations, vous retrouvez la **variable** *somme\_argent* (c'est ainsi que j'ai décidé de finement la nommer), qu'il faudra **initialiser** avec la valeur de l'argent que j'ai compté. C'est une **donnée d'entrée**. Dans l'exemple scratch on l'initialise avec une **saisie** par l'opérateur (« demander .. »), on y reviendra.

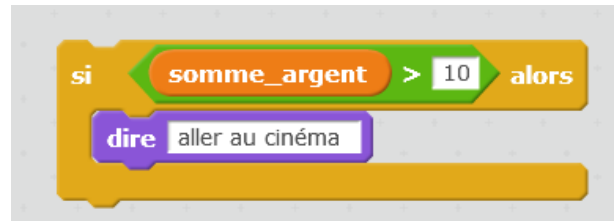
## 2. Les instructions conditionnelles

Comme indiqué dans son nom une telle instruction contient une condition introduite par « si ».

**Si** la condition est remplie (autrement dit, si la condition est **vraie**) **alors** je fais ceci.

Exemple :

si j'ai plus de 10 €  
alors je vais au cinéma.



On peut compliquer avec **sinon**.

Si la condition est remplie (si la condition est **vraie**)  
alors je fais ceci,  
**sinon** (si la condition est **fausse**) je fais cela.

Nous l'avons déjà utilisée dans le premier exemple au dessus.

En version anglaise (it could be usefull) on utilise if, then, else :



### 3. Les variables

Nous avons eu un 1<sup>er</sup> exemple de variable plus haut : *somme\_argent*

Une **variable**, est une zone mémoire à laquelle le programmeur donne le nom qu'il veut (ou presque!). Elle peut alors prendre n'importe quelle valeur.

Une variable est bien pratique, et même indispensable, pour faire des calculs et ou mémoriser des données d'entrée ou des résultats.

*Exemple : la famille Ronaldo achète un terrain rectangulaire dont elle connaît la longueur et la largeur, et elle veut connaître sa surface pour savoir si on peut y faire un terrain de football (tiens, pourquoi ?)*

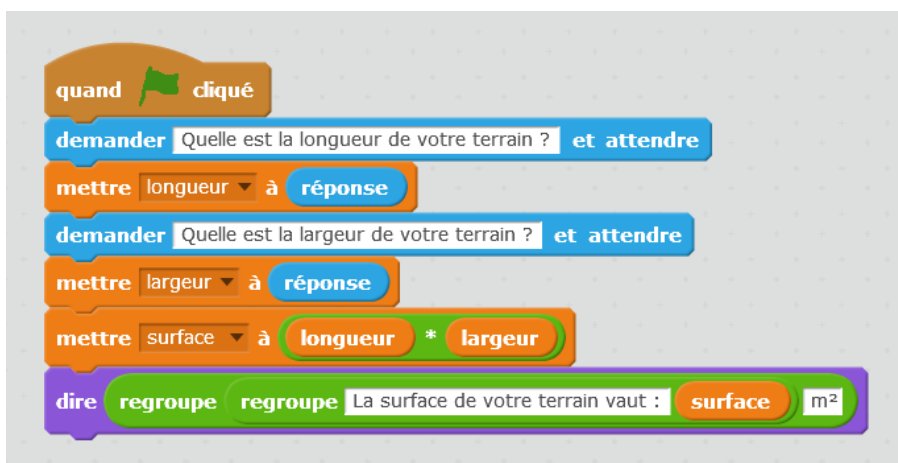
On a donc ici deux **données d'entrée**, qu'on va demander à l'utilisateur de saisir : la longueur et la largeur. On va donc créer 2 variables qu'on peut judicieusement appeler **longueur** et **largeur**.

Le programme va ensuite effectuer la multiplication :  $\text{surface} = \text{longueur} \times \text{largeur}$   
On va donc créer en plus une variable **surface** qui sera par conséquent une **donnée de sortie**, (c'est à dire un résultat).

On met le résultat dans cette variable :  $\text{surface} \leftarrow \text{longueur} * \text{largeur}$

(souvenez-vous que dans tous les langages de programmation, la multiplication s'écrit à l'aide de l'astérisque \* et la division avec le slash /).

On affiche ensuite le résultat. Ce qui donne :



Le résultat à l'écran :



## 4. Les boucles

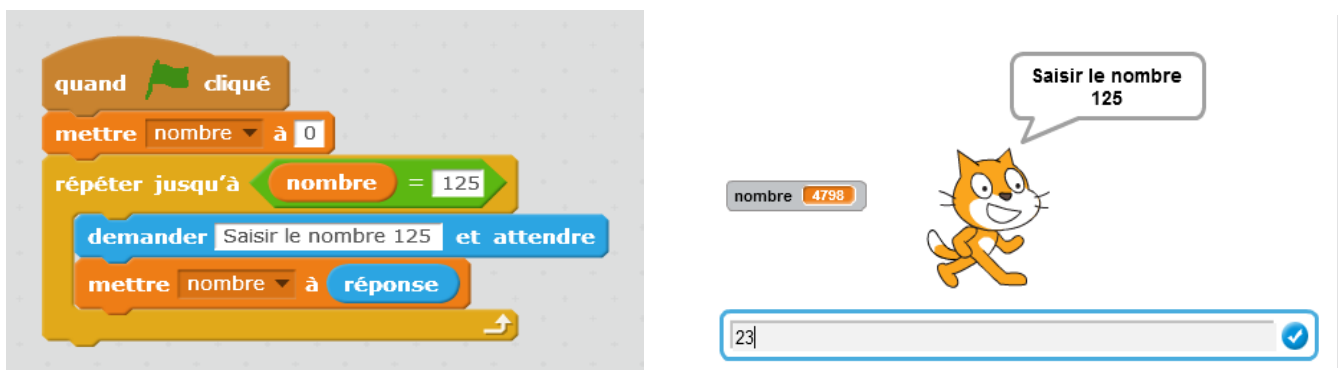
Les boucles permettent de répéter autant de fois que nécessaire la même action :

### La boucle *répéter jusqu'à* :

Ce type de boucle s'exécute ***tant que*** la condition qui nous intéresse n'est pas vraie.

Exemple : *je veux que l'utilisateur saisisse le nombre 125. Je lui redemande tant qu'il ne rentre pas la bonne valeur.*

Cela donne :



The image shows a Scratch script on the left and its execution on the right. The script starts with a 'when clicked' event, followed by 'set number to 0'. A 'repeat until' loop contains a 'say "Saisir le nombre 125" and wait' block and a 'set number to response' block. On the right, the Scratch cat is shown with a speech bubble saying 'Saisir le nombre 125'. A variable monitor shows 'nombre' with the value 4798. Below the cat is a text input field containing '23' and a checkmark button.

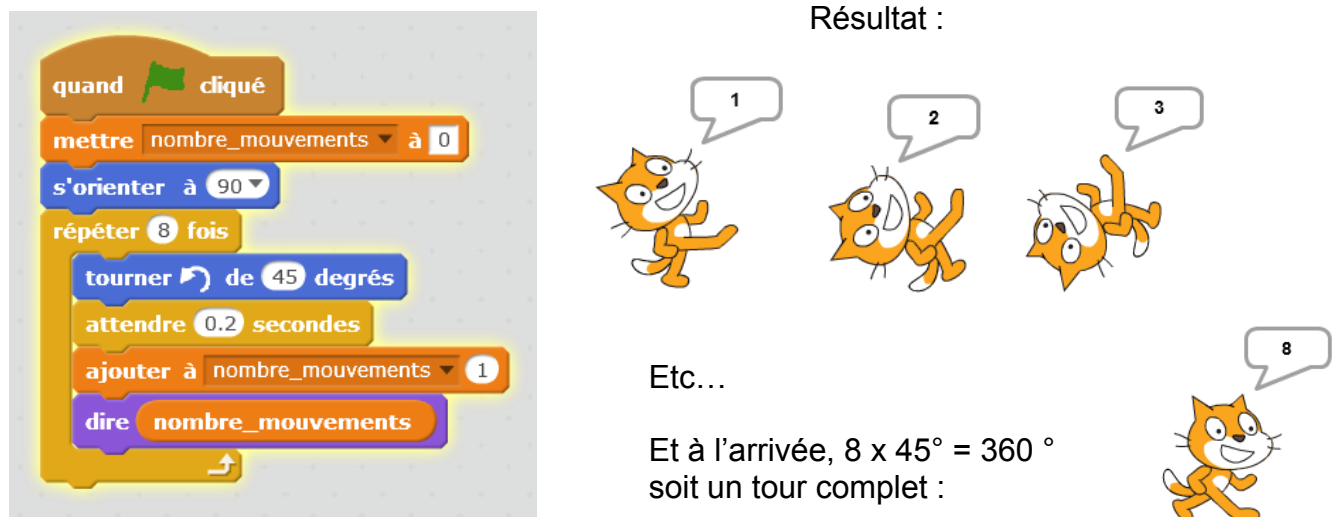
La saisie est mémorisée ici dans la variable *nombre*. Le programme demande de saisir le nombre jusqu'à ce que celui-ci égale 125.

### La boucle *répéter plusieurs fois* :

On peut définir le nombre de fois que l'on veut répéter une action.

Exemple : *je fais faire un tour complet de notre chat préféré par pas de 45° et à chaque fois je lui demande de compter le nombre de pas (de mouvements).*

Ici aussi on a besoin d'une variable que je vous laisse le soin d'identifier. Cela donne :



The image shows a Scratch script on the left and its result on the right. The script starts with a 'when clicked' event, followed by 'set number\_mouvements to 0', 'turn 90 degrees', and a 'repeat 8 times' loop. Inside the loop are 'turn 45 degrees', 'wait 0.2 seconds', 'add 1 to number\_mouvements', and 'say number\_mouvements'. On the right, the text 'Résultat :' is followed by three illustrations of the Scratch cat in different orientations, each with a speech bubble containing the number 1, 2, and 3 respectively. Below these is 'Etc...' and another illustration of the cat with a speech bubble containing the number 8. At the bottom, the text reads 'Et à l'arrivée,  $8 \times 45^\circ = 360^\circ$  soit un tour complet :