

Systeme d'arrosage automatique

22 – Programmation partie 5

Travail sur la partie "processeur" du système d'arrosage automatique (programmation). Suite du travail sur les programmes.

Durée : entre 10 et 15 minutes.

Le travail s'effectuera encore sur Scratch. Tous les exemples utiliseront donc la notation Scratch.

Le travail de cette semaine concerne à nouveau les boucles.

Plusieurs d'entre vous ont eu du mal avec cette partie la semaine dernière, je vais donc tenter de l'expliquer autrement.

4 – Boucles (suite)

Les boucles sont un mécanisme qui permet de répéter plusieurs fois de suite une partie d'un programme.

C'est un mécanisme un peu particulier dont on ne dispose pas naturellement mais que l'on trouve tout de même dans la vie de tous les jours.

Naturellement, si on veut dire dix fois "bonjour", nous sommes obligés de répéter dix fois le mot "bonjour". Mais on peut aussi utiliser un magnétophone, enregistrer "Bonjour", puis passer l'enregistrement 10 fois "en boucle".

On entendra 10 fois bonjour, mais on ne l'aura prononcé qu'une fois.

Pour un programme, le principe est le même.

Pour exécuter plusieurs fois la même chose le programmeur a deux solutions :

- Soit il écrit la même chose autant de fois que nécessaire, ce qui est déjà compliqué si il connaît le nombre de répétitions, mais qui est impossible si il ne connaît pas à l'avance le nombre de répétitions nécessaires.
- Soit il utilise une boucle.

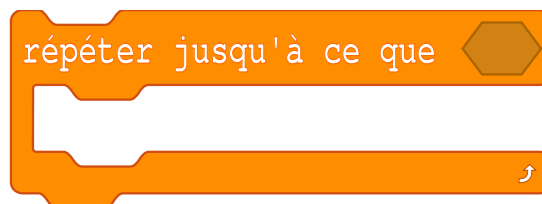
C'est à ça que servent les boucles.

Elles permettent de répéter une opération (ou un groupe d'opérations) un certain nombre de fois.

Il y a trois types de boucles en Scratch, selon le besoin :

- celles qui permettent de répéter les opérations un nombre donné de fois
- celles qui permettent de répéter les opérations jusqu'à ce qu'une condition soit remplie
- celles qui permettent de répéter les opérations indéfiniment

En Scratch, les boucles s'écrivent à l'aide de l'un des "Contrôle" suivants :

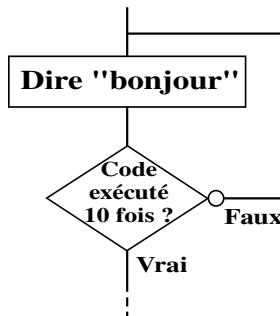


Cas 1

Le premier cas "répéter un nombre donné de fois" correspond à mon exemple : le premier bonjour "enregistré" est celui que le programmeur écrit à l'intérieur de la boucle. Ce qui donne ceci :



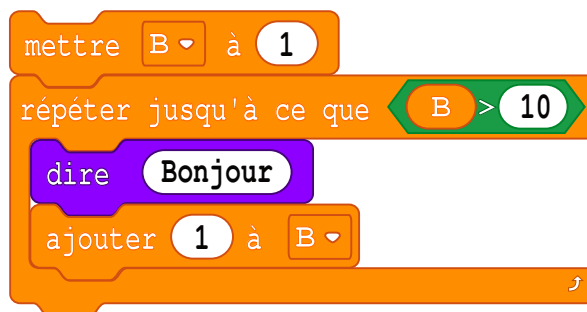
et qui correspond à l'organigramme suivant :



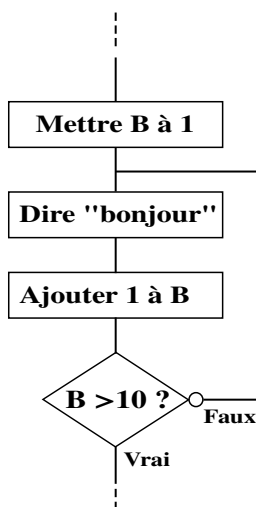
On remarque qu'avec la notation Scratch, le test "Code exécuté 10 fois ?" est "masqué".

Cas 2

Le second cas "répéter les opérations jusqu'à ce qu'une condition soit remplie" est le cas le plus général. Pour obtenir le même résultat qu'avec le cas 1 (dire 10 fois bonjour) il faut utiliser une variable intermédiaire que l'on appelle alors un "compteur", ce qui donne ceci :

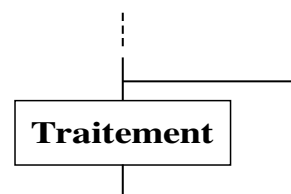


et qui correspond à l'organigramme suivant :



Cas 3

Et enfin, le troisième cas est un cas particulier, qui n'a pas de condition de sortie ! Le programme reste alors bloqué dans la boucle, formant ce que l'on appelle une boucle infinie !



Ce type de boucle ne permet pas de dire bonjour seulement 10 fois, et ne peut donc pas être utilisé pour notre exemple.

Il est tout de même utile dans de nombreux cas, comme celui de la fiche 18, mais aussi dans de très nombreux programmes qui fonctionnent "en boucle" de façon permanente. C'est le cas de la majorité des systèmes embarqués, qui remplissent leur fonction "en boucle" tant qu'ils sont allumés, et n'ont pas de raison de s'arrêter. Leur programme utilise une boucle infinie.

Voilà, j'espère que c'est un peu plus clair maintenant.