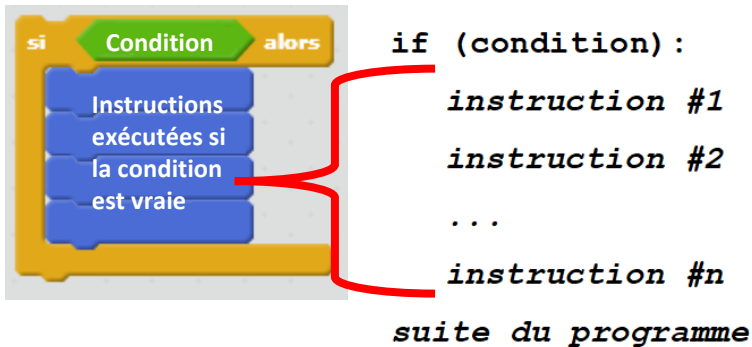


Parallèle Scratch / Python

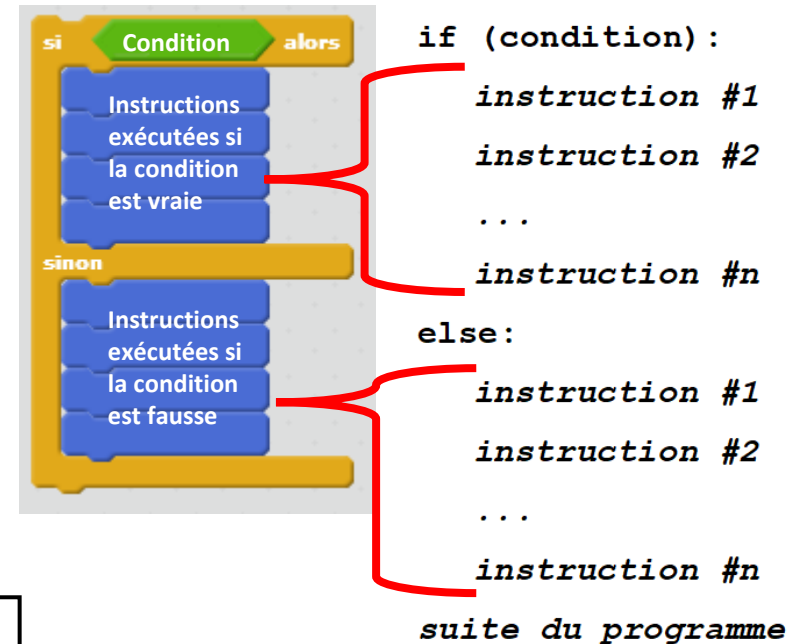
Instructions conditionnelles



Test « si – alors »



Test « si – alors – sinon »



Conditions :

	<code>var1 < var2</code>
	<code>var1 == var2</code>
	<code>var1 > var2</code>
	<code>(condition #1) and (condition #2)</code>
	<code>(condition #1) or (condition #2)</code>
	<code>not(condition #1)</code>

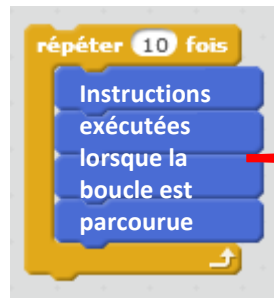
Conditions supplémentaires

En Python :

<code>var1 <= var2</code>	<code>var1 ≤ var2</code>
<code>var1 != var2</code>	<code>var1 ≠ var2</code>
<code>var1 >= var2</code>	<code>var1 ≥ var2</code>

Parallèle Scratch / Python

Boucles bornées



`for i in range(10):` la boucle est parcourue 10 fois

`instruction #1` (i varie de 0 à 9)

`instruction #2`

`...`

`instruction #n`

`suite du programme`

En plus en Python :

`for i in range(a, b):` i varie de a (inclus) à b (exclus)

`instruction #1`

`instruction #2`

`...`

`instruction #n`

`suite du programme`

`for i in range(a, b, c):` i varie de a (inclus) à b (exclus)

`instruction #1`

à chaque tour, on ajoute c à i

`instruction #2`

(c peut être négatif pour compter

`...`

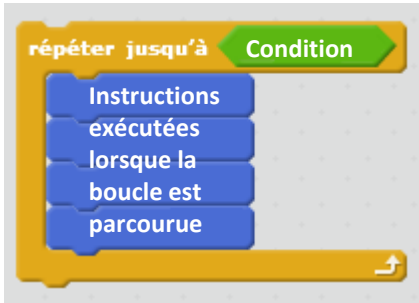
à rebours)

`instruction #n`

`suite du programme`

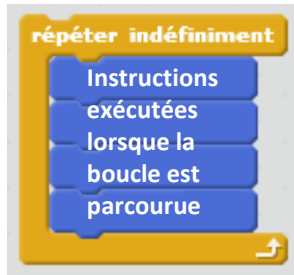
Parallèle Scratch / Python

Boucles non bornées



```
while (condition):  
    instruction #1  
    ...  
    instruction #n  
suite du programme
```

La boucle est parcourue tant que la condition dans le **while** est vraie, contrairement au Scratch où la boucle est parcourue jusqu'à ce que la condition soit vraie. Si la condition en Scratch est $a = b$ par exemple, il faudra utiliser $a \neq b$ en Python.



```
while (True):  
    instruction #1  
    instruction #2  
    ...  
    instruction #n  
suite du programme
```

La boucle est parcourue tant que la condition dans le **while** est vraie, donc indéfiniment dans ce cas. (On pourrait remplacer **True** par une condition toujours vraie, comme $2 == 2$)

En plus en Python :

On peut sortir prématurément d'une boucle bornée ou non bornée à l'aide de l'instruction **break**. La poursuite du programme se fait à partir de la première instruction suivant la boucle. De même qu'en Scratch, on peut imbriquer plusieurs boucles.