
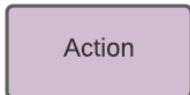
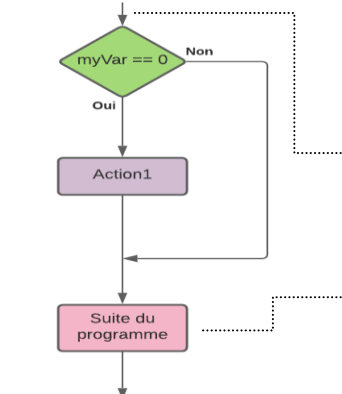
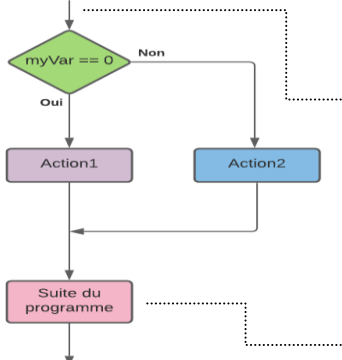
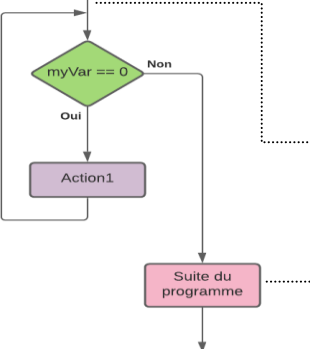
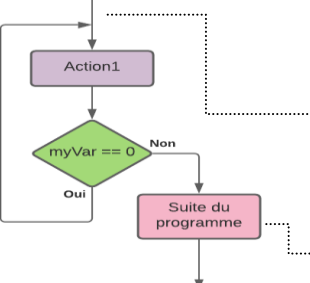
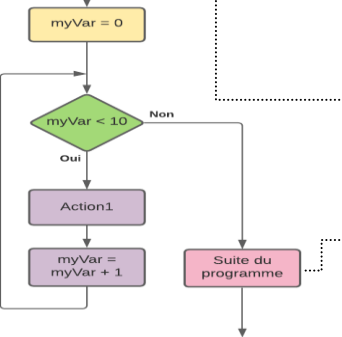
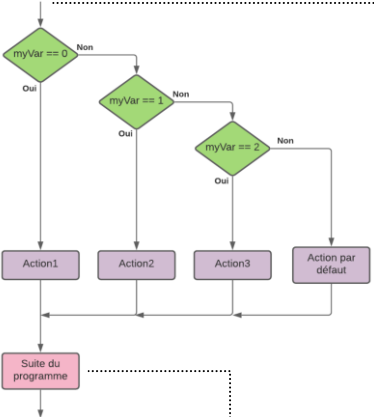


Correspondance algorithme vers langage Arduino

Symboles	Langage Arduino								
	Condition correspond à une <u>expression</u> , par exemple :								
	<code>myVar == 0 // la variable myVar contient-elle 0 ?</code>								
	ou								
	<code>myVar != 10 // la variable myVar a-t-elle une valeur différente de 10 ?</code>								
	ou								
	<code>myVar < 10 // la variable myVar est-elle inférieure à 10 ?</code>								
	On peut également utiliser des opérateurs logiques :								
	<table><tr><th>Nom</th><th>Langage Arduino</th></tr><tr><td>ET</td><td>&&</td></tr><tr><td>OU</td><td> </td></tr><tr><td>NON</td><td>!</td></tr></table>	Nom	Langage Arduino	ET	&&	OU		NON	!
Nom	Langage Arduino								
ET	&&								
OU									
NON	!								
	Exemple :								
	<code>(myVar >= 5) && (myVar <= 10) //myVar a-t-elle une valeur comprise entre 5 et 10 inclus ?</code>								
	Action correspond à une ou plusieurs instructions, par exemple :								
	<code>digitalWrite(13, LOW); delay(500); digitalWrite(13, HIGH); delay(500);</code>								

Nom de la structure de contrôle	Algorithme	Code Arduino (langage de programmation C)	Remarque
SI (alternative simple)		<pre>boolean myVar; myVar = 0; // Début du SI if (myVar == 0) { // Action1 } // Suite du programme</pre>	myVar == 0 peut être remplacé par n'importe quelle expression.
SI - SINON (alternative complète)		<pre>boolean myVar; myVar = 0; // Début du SI - SINON if (myVar == 0) { // Action1 } else { // Action2 } // Suite du programme</pre>	myVar == 0 peut être remplacé par n'importe quelle expression.

<p>TANT QUE FAIRE (répétition conditionnelle)</p>		<pre>boolean myVar; myVar = 0; // Début de la boucle TANT QUE FAIRE while (myVar == 0) { // Action1 } // Suite du programme</pre>	<p>myVar == 0 peut être remplacé par n'importe quelle expression.</p>
<p>FAIRE TANT QUE (répétition conditionnelle)</p>		<pre>boolean myVar; myVar = 0; // Début de la boucle FAIRE TANT QUE do { // Action1 } while (myVar == 0); // Suite du programme</pre>	<p>Action1 est exécutée au moins une fois ! Attention à bien mettre le point-virgule après le while !</p>
<p>POUR (répétition conditionnelle)</p>		<pre>// Début de la boucle POUR for (int myVar = 0; myVar < 10; myVar++) { // Action1 } // Suite du programme</pre>	<p>Action1 est exécutée 10 fois dans cet exemple.</p>
<p>CHOIX MULTIPLE</p>		<pre>int myVar; // Début du choix multiple switch (myVar) { case 0 : // Action1 break; case 1 : // Action2 break; case 3 : // Action3 break; default : // Action par défaut } // Suite du programme</pre>	<p>S'il n'y a pas d'action par défaut la clause default peut être supprimée !</p>

Pour plus d'informations sur le langage Arduino, consultez le site suivant :

<https://docs.arduino.cc/language-reference/>