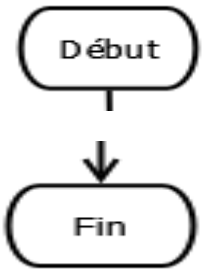
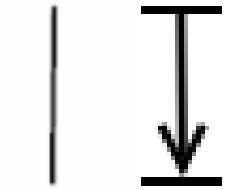
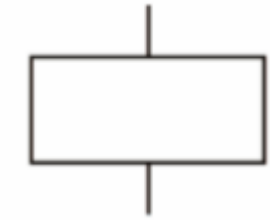
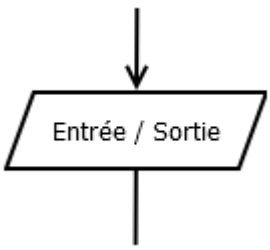
S si	**Algorigramme / organigramme / logigramme**	**Cours**

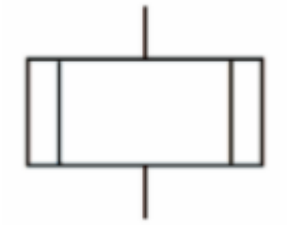
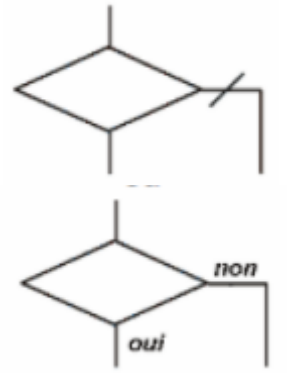
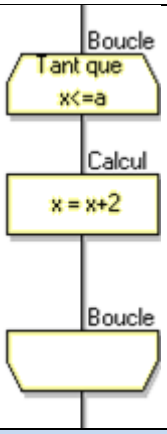
1. Fonction

Le rôle d'un algorigramme est de représenter un algorithme.
La représentation graphique utilisée permet d'en simplifier la lecture.
Les algorigrammes ne peuvent être utilisés que pour représenter des programmes relativement cours

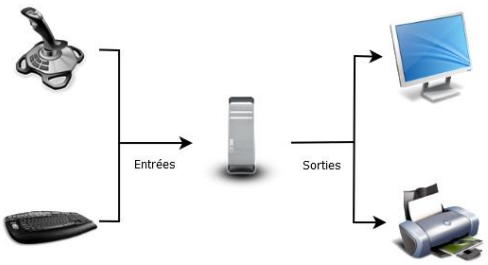
2. Symboles utilisés

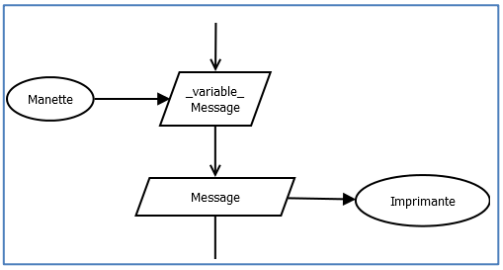
Pour construire un algorigramme, on utilise des symboles normalisés

			
Début / fin Il n'y a qu'un seul « Début » mais il peut y avoir plusieurs « Fin »	Liaison, avec ou sans flèche, permettent de guider la lecture de l'algorigramme	Instructions : elles permettent de réaliser les traitement : calculs, comparaisons, …	Lecture ou écriture d'une donnée

			
Sous-programme	Test	Boucle	Commentaire

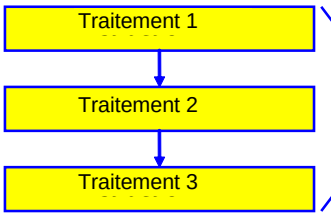
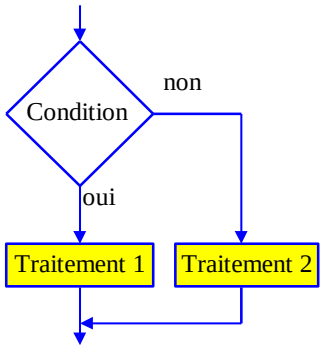
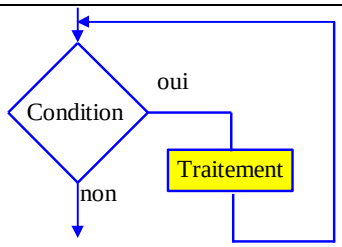
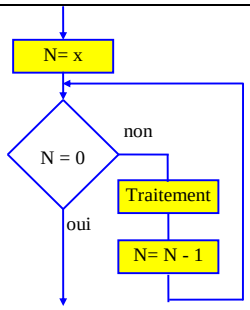
Les périphériques. Ils sont représentés dans des formes ovales. Une flèche indique s'ils fournissent une valeur en entrée ou s'ils constituent une sortie.
Les entrées sont généralement mises à gauche et les entrées à droite.





3. Structures

Exemples de structures :

Structure	Algorigramme	Algorithme
Structure linéaire La structure linéaire se caractérise par une suite d'actions à exécuter successivement dans l'ordre de leur énoncé.		**FAIRE « traitement 1 »** **FAIRE « traitement 2 »** **FAIRE « traitement 3 »**
Structure alternative ou conditionnelle Une structure alternative n'offre que deux issues possibles s'excluant mutuellement. Une condition est testée et en fonction du résultat du test soit le traitement 1, soit le traitement 2 est réalisé.		**SI « condition » vraie** **ALORS FAIRE** **« traitement 1 »** **SINON FAIRE** **« traitement 2 »** **FIN SI**
Structure répétitive ou itérative (boucle avec pré-test) Dans cette structure on commence par tester la condition, si elle est vraie alors le traitement est exécuté.		**TANT QUE « condition » vraie** **FAIRE « traitement »** **FIN TANT QUE**
Boucle avec comptage On initialise la variable N avec une valeur x. On teste si N est égal à 0, si ce n'est pas le cas, on exécute le traitement et on décrémente la variable N puis on teste à nouveau la variable N, et ainsi de suite jusqu'à ce que N=0.		**POUR N = x A 0 REPETER** **« traitement »** **FIN POUR**