

Pratique algorithmique

1.1- La notion de variable en informatique

Une **variable** informatique 'x' est comme une étiquette collée sur une boîte qui peut contenir différentes valeurs. Le contenu de la boîte :

- varie au cours du déroulement de l'algorithme (ou de l'exécution du programme correspondant) ;
- peut être de nature différente (nombre, caractères...) selon la variable.

Exemple : la variable `mon_age` contient un nombre (qui peut changer en fonction de l'algorithme). La variable `mon_texte` contient une chaîne de caractères (et peut aussi changer dans l'algorithme)

`mon_age = 53`

`mon_texte = « Bonjour »`

Si dans un algorithme ou un programme vous trouvez l'expression $x+4$, elle sera évaluée (calculée par l'ordinateur) comme la somme du nombre contenu dans la boîte étiquetée x et de 4.

Une instruction d'affectation comme $y = x + 4$ signifie :

- évaluer l'expression $x + 4$, en ajoutant 4 à la valeur contenue dans la boîte étiquetée x ;
- mettre le résultat dans la boîte étiquetée y .

Enfin, sachez que lorsqu'on crée une variable dans un programme, une valeur par défaut lui est attribuée. Pour des variables destinées à contenir des valeurs numériques, cette valeur est de 0. Pour des variables destinées à contenir des chaînes de caractères elle correspond à une chaîne de caractères vide.

1.2- Déroulement de l'exécution d'un algorithme

Les algorithmes sur lesquels nous allons travailler peuvent être vus comme une recette de cuisine. Pour une recette, nous décrivons les instructions pas à pas et dans l'ordre dans lequel elles doivent être réalisées.

Pour un algorithme ce sera exactement la même chose : nous devons décrire nos algorithmes en décrivant les instructions dans l'ordre dans lequel nous voulons qu'elles soient exécutées par l'ordinateur (lorsque l'algorithme sera traduit en programme informatique).

Deux instructions viennent modifier le déroulement en séquence d'un algorithme : les instructions conditionnelles et les boucles. Ce sont ce qu'on appelle en informatique les instructions de contrôle.

Instructions de contrôle	Instruction conditionnelle	Boucle	
Pseudo code : description indépendante du langage de programmation et compréhensible par les développeurs	SI condition ALORS suite_d'instructionsA SINON suite_d'instructionsB La partie SINON est optionnelle.	REPETER compteur FOIS suite_d'instruc tions	TANT QUE condition ALORS suite_d'instruc tions
Que fait l'instruction ?	Si condition est vraie, alors suite_d'instructionsA sera exécutée par l'ordinateur, sinon ce sera suite_d'instructionsB.	Répéter x fois la suite d'instruction. → <i>Équivalent de la boucle for on y reviendra</i>	Tant que condition est vraie, alors suite_d'instructions sera exécutée par l'ordinateur.

1.3- Pour s'entraîner

Pour vous entraîner à décrire les algorithmes que vous avez dû utiliser dans Algoréa, vous pouvez reprendre certains défis Algoréa et pour les défis vous devez :

1. comprendre le problème en exécutant dans sa tête ce qui est demandé ;
2. écrire l'algorithme correspondant en français et sur papier (pseudo code) ;
3. Faire ou refaire le programme d'après votre algorithme
4. décrire le comportement du programme réalisé sur Algoréa

Vous pourrez utiliser selon les défis :

- Des instructions de boucle (TANT QUE...ALORS ou REPETEZ x fois)
- Des instructions conditionnelles (SI...ALORS...SINON)
- Des variables