



TD n°2

Objets et Actions élémentaires -
Structures de contrôle conditionnelles

Résumé de cours :

Une **variable** est un élément dont la valeur peut évoluer au fil d'un algorithme. Elle est désignée par un nom significatif, son type décrit l'utilisation possible qui peut en être faite. Par convention, le nom commence par une minuscule. Lorsqu'il est composé de deux mots, le deuxième mot commence par une Majuscule sans espace exemple : **indiceCourant**.

Nom de la variable	carLu	Nom de la variable	valNum
Adresse en mémoire	Adr. 0001	Adresse en mémoire	Adr. 1025
Représentation graphique	A	Valeur en décimal	118
Valeur ASCII (décimal)	65	Valeur Binaire	0 1 1 1 0 1 1 0
Valeur ASCII (hexadécimal)	41		
Valeur Binaire	0 1 0 0 0 0 0 1		1 octet

1 octet

L'**affectation** a pour rôle de donner une valeur à une variable.

Exemple : **variable** ← **expression** la variable reçoit le résultat de l'expression. **valNum** ← **118**



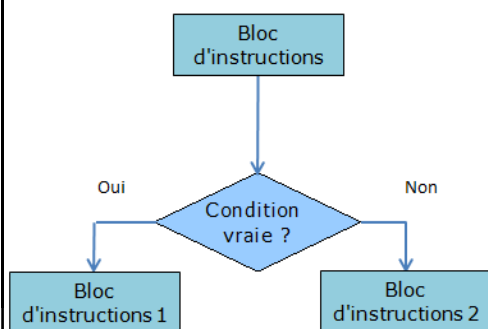
L'action **Lire : variableLue**

Lit la valeur saisie au clavier par l'utilisateur et la place dans la variable variableLue.



L'action **Ecrire : variableEcrire**

Fais apparaître sur l'écran le contenu de la variable variableEcrire



Le **schéma conditionnel** Si...alors...sinon.... permet une alternative entre deux séquences.

Si la condition est vraie
alors Bloc Instructions 1
sinon Bloc Instructions 2

FinSi

Le **sinon** est facultatif.

Pour chacun des algorithmes vous utiliserez, les structures de contrôles suivantes : Début, Fin, faire... tant que...

Exercice n° 1 : Comprendre une séquence algorithmique.

Soit le schéma algorithmique suivant :

début

écrire : "entrez un nombre positif"

lire : nombreLu

coefficient $\leftarrow 2 * \text{nombreLu} - 1$

Somme $\leftarrow 0$

faire

 somme \leftarrow somme + coefficient

 coefficient \leftarrow coefficient - 2

tant que coefficient ≥ 1

écrire : somme

fin

- Décrivez l'environnement et le lexique de cette séquence. En simulant le comportement de la séquence pour plusieurs valeurs de **nombreLu**, indiquez le contenu de la variable **somme** à la fin de la séquence. Que pouvez-vous conclure de cet algorithme ?
- Comment faut-il modifier l'itération pour conserver le même résultat si l'on remplace l'action somme $\leftarrow 0$ par somme \leftarrow coefficient ?
- Récrivez une autre séquence qui effectue le même calcul, mais en faisant varier Coefficient par valeurs croissantes.

Exercice n° 2 : Décomposer une action complexe en actions primitives.

En utilisant l'idée illustrée en Annexe, réalisez un algorithme qui permet à un utilisateur de saisir un entier et d'afficher sa valeur correspondante en binaire. On attachera une attention particulière à la description de l'environnement, du lexique et dans le choix des noms de variables.

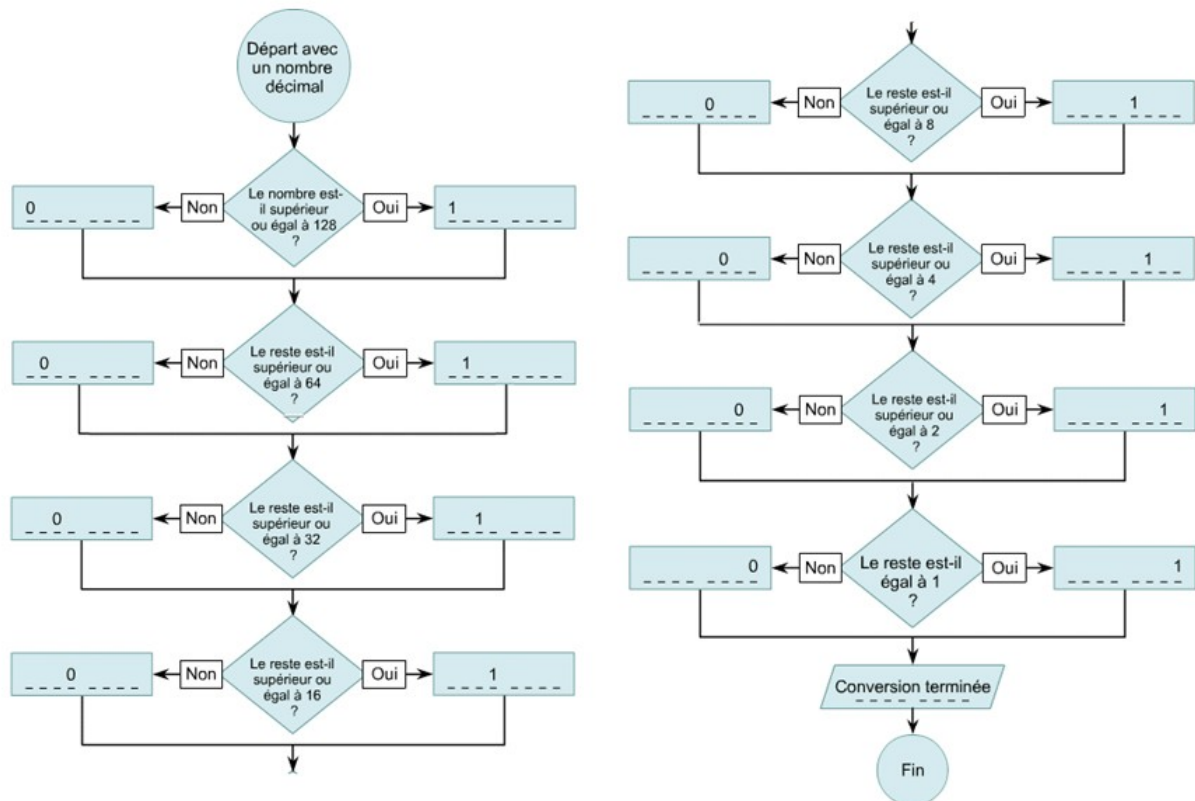
Exercice n° 3 : Établir une séquence à l'aide d'actions primitives.

En utilisant l'idée illustrée en annexe, réalisez un algorithme qui permet à un utilisateur de saisir un à un les bits d'un octet et d'afficher sa valeur correspondante en décimale. On attachera une attention particulière à la description de l'environnement, du lexique et dans le choix des noms de variables.

Dans le contexte de cet algorithme, de quels éléments serait constitué le lexique de variable ?

Annexes :

Conversion décimale binaire



Conversion d'un nombre binaire sur 8 bits en décimal

