

Série Je Programme

Préparé par :
H.H

CONCOURS ALGORITHMIQUE

```
graph TD; B1[ ] --> B2[Algorithme Addition]; B2 --> B3["begin A,b,c : entiers"];
```

+ Plusieurs Exercices...

**c = a + b;
ecrire(c);
fin.**

EDITION E.P.E.R.S

SOMMAIRE

PARTIE 01 : COURS

1. INTRODUCTION	01
2. REPRESENTATION GRAPHIQUE DE L'ALGORITHME	03
3 . STRUCTURE D'UN ALGORITHME	03
4. LES OPERATEURS	06
5. LES STRUCTURES ALGORITHMIQUES FONDAMENTALES	08
6.FORME GENERALE D'UN ALGORITHME	15
7. DEMARCHE GENERALE	17
8 . INSTRUCTION DE BASE	17
9. LES EXPRESSIONS	21
10. L'INSTRUCTION CONDITIONNELLE « SI »	23
11. LES INSTRUCTIONS REPETITIVES	27
12. LES TABLEAUX	36

PARTIE 02 : EXERCICES

1. INTRODUCTION :

Cette marche à suivre porte le nom d'algorithme qui provient du nom d'un célèbre mathématicien arabe de la première moitié du IXe siècle:

Muhammad ibn Musa al Khwarizmi.

Algorithme

C'est un ensemble de règles opératoires rigoureuses, ordonnant à un processeur d'exécuter dans un ordre déterminé un nombre fini d'opérations élémentaires ; il oblige à une programmation structurée.

Ce langage utilise un ensemble de mots clés et de structures permettant de décrire de manière complète, claire, l'ensemble des opérations à exécuter sur données pour obtenir des résultats ; on n'hésitera donc pas à agrémenter l'algorithme de nombreux commentaires.

Un algorithme est écrit en utilisant un langage de description d'algorithme (LDA) .

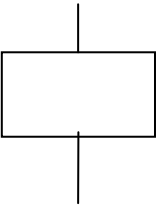

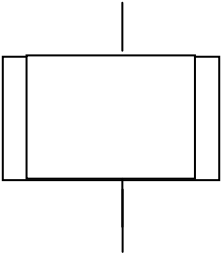
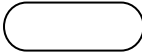
L'algorithme ne doit pas être confondu avec le programme proprement dit.

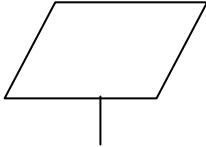
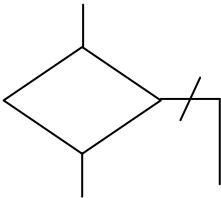
2.REPRÉSENTATION GRAPHIQUE DE L'ALGORITHME

Algorithme :

Pour le construire, on utilise des symboles normalisés.

Quelques symboles utilisés dans la construction d'un algorithme :

SYMBOLE	DESIGNATION	SYMBOLE	DESIGNATION
Symboles de traitement		Symboles auxiliaires	
	Symbole général Opération ou groupe d'opération sur des données, instructions, pour laquelle il n'existe aucun symbole normalisé.		Renvoi Symbole utilisé deux fois pour assurer la continuité lorsque une partie de ligne de liaison n'est pas représentée.
	Sous programme Portion de programme considérée comme une simple opération.		Début, fin, interruption, Début, fin ou interruption d'un algorithme.

	Entrée. Sortie Mise à disposition d'une information à traiter ou enregistrement d'une information traitée.		Commentaire Symbole utilisé pour donner des indications sur les opérations effectuées
Symbole de test		Les différents symboles sont reliés entre eux par des lignes de liaisons.	
	Branchement Exploitation de conditions variables impliquant un choix parmi plusieurs.		
Sans conventionnel des liaisons			
<p>Note 1 . Le sens général des lignes de liason doit être ;</p> <ul style="list-style-type: none"> ● De haut en bas ● De gauche à droite <p>Lorsque le sens général ne peut pas être respecté, des pointes, de flèche à cheval sur la ligne indiquent le sens utilisé.1</p>			

Note :

L'avantage d'un langage d'algorithme est de pouvoir être facilement transcrit dans un langage de programmation structuré (pascal, c, c ++, ...)

3 – Structure d'un algorithme :

Algorithme nom de l'algorithme ;

CONST Liste des constantes ;

Les déclarations de constante, variables, structures

VAR Liste des variables ;

STRUC Liste des structures

FONC Liste des fonctions

Les déclarations de fonctions et procédures

PROC Liste des procédures

DEBUT

Le corps de l'algorithme

Action 1 ;

Action 2 ;

–

–

Action n ;

FIN ALGORITHME

Tous les mots clés sont soulignés et écrits en **Gras** MAJISCULE. Une marque de terminaison (;) est **utilisée** entre chaque action.

4- Les opérateurs :

a. Opérateurs sur les entiers et les réels

Arithmétiques	
+	Addition
-	Soustraction
*	Multiplication
/	Division
DIV	Division entière
↑	puissance

Comparaisons	
>	Supérieur
<	Inférieur
≥	Supérieur ou
≤	égal
=	Inférieur ou
≠	égal
	Egal
	Différent

b. Opérateurs sur les entiers et les booléens

Fonction logique	
Et	Fonction ET
Ou	Fonction OU
Oux	Fonction OU
Non	exclusif
Non et	Fonction NON
Non ou	Fonction NON ET
>>	Fonction NON OU
<<	Décalage à droite
	Décalage à gauche

Fonction de comparaison	
=	Egal
≠	différent

c – Opérateurs sur les caractères et les chaînes

Fonction de Concaténation	
+	concaténation

Fonction de comparaison pour les chaînes	
=	Egalité
≠	Différent
>	Supérieur
<	Inférieur

d – Priorité des opérateurs

Priorité à la multiplication et à la division.

e - L'affectation

Elle permet d'affecter une valeur à une variable.

Syntaxe : identificateur de la variable ←
expression

L'expression est une suite d'opérations sur des constantes ou des variables déjà déclarées.

5- Les structures algorithmiques fondamentales :

Les opérations élémentaires relatives à la résolution d'un

problème peuvent, en fonction de leur enchaînement être organisées suivant quatre familles de structures algorithmiques fondamentales.

- Structure linéaire
- Structure alternatives
- Structure de choix
- Structure itératives (ou répétitives)

a – Caractéristique de la structure linéaire

La structure linéaire se caractérise par une suite d'actions à exécuter successivement dans l'ordre énoncé.

Notation : Faire action ;

b - Caractéristique de la structure alternative

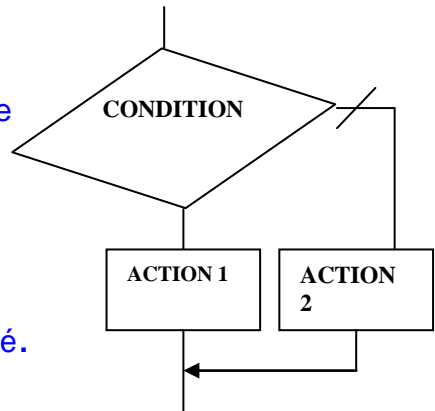
La structure alternative n'offre que deux issues possibles à la poursuite de l'algorithme et s'excluant mutuellement.

On peut rencontrer deux types de structures alternatives :

b.1. Une structure alternative complète

Dans cette structure l'exécution d'un des deux traitements distincts ne dépend que du résultat d'un test effectué sur la condition qui peut être une variable ou un évènement ;

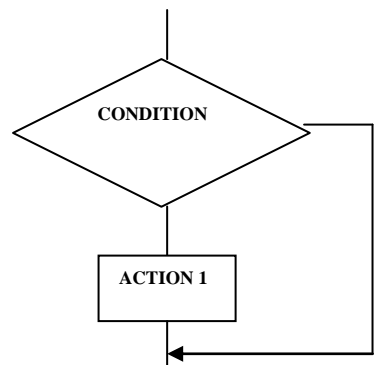
Si la condition est vérifiée seul le premier traitement est exécuté ;
si la condition n'est pas vérifiée seul le second traitement est effectué.



Notation : Si condition <u>alors</u> Action 1 <u>Sinon</u> Action 2 ; <u>Fsi</u> ;
--

b.2 – Une structure alternative réduite

La structure alternative réduite se distingue de la précédente par le fait que seule la situation correspondant à la validation de la condition entraîne l'exécution du traitement, l'autre situation conduit systématiquement à la sortie de la structure.

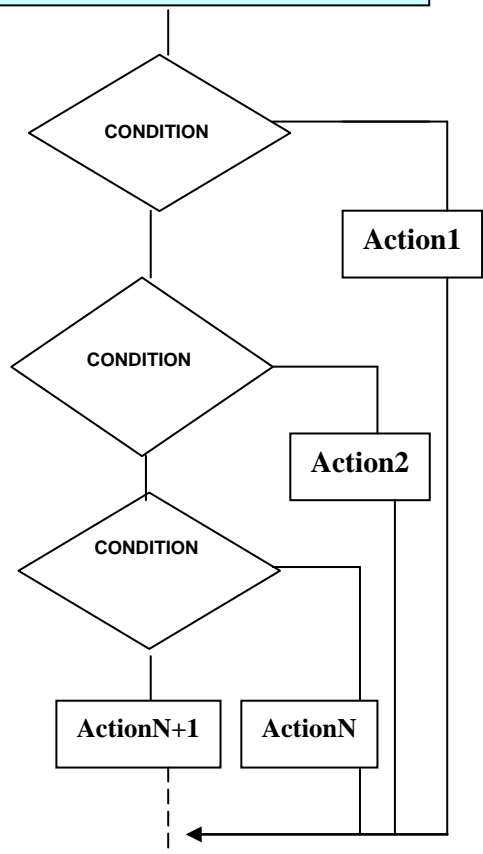


Notation : Si condition alors
 Action ;
 Fsi ;

c- Caractéristique de la structure de choix

La structure de choix permet, en fonction de plusieurs conditions de type booléen, d'effectuer des actions différentes suivant les valeurs que peut prendre une même variable.

Notation : suivant valeur faire
 Valeur 1 : action 1 ;
 Valeur 2 : action 2 ;
 :
 :
 Valeur N : action N ;
 Sinon action N+1 ;
 Fsuivant ;



d – Caractéristique de la structure itérative

La structure itérative répète l'exécution d'une opération ou d'un traitement

On considérera 2 cas :

Premier cas : le nombre de répétitions n'est pas connu ou il est variable.

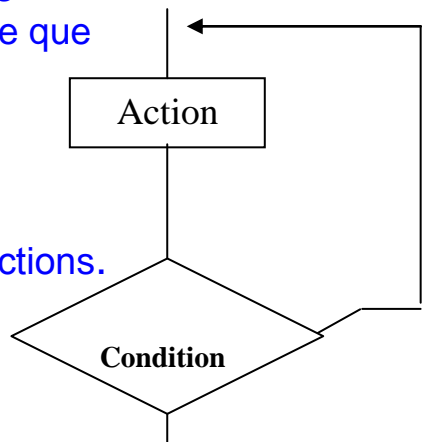
On distingue 2 structures de base :

Structure **RÉPÉTER JUSQU'À**

Dans cette structure, le traitement est exécuté une première fois puis sa répétition se poursuit jusqu'à ce que la condition soit vérifiée.

Par traitement on entend :

- soit une structure isolée ;
- soit une succession d'instructions.



L'action est toujours exécutée au moins une fois.