



ANNÉE SCOLAIRE 2023/2024

COURS D'ALGORITHMIQUE

Pape Abdoulaye **BARRO**

Docteur en Informatique et Télécommunications

Spécialiste en Télémétrie & Systèmes Intelligents

ALGORITHMIQUE

PLAN

- + Généralité & bases
- + Structures conditionnelles
- + Structures itératives
- + Tableaux
- + Sous programmes

ALGORITHMIQUE

PLAN > GÉNÉRALITÉ & BASES

- + Objectifs
- + Preamble
- + Notion d'algorithme
- + Types et variables
- + Lecture et écriture
- + Operateurs, conditions et expression

ALGORITHMIQUE GÉNÉRALITÉ > OBJECTIFS

L'objectif de ce cours est de savoir transcrire les différentes étapes de résolution d'un problème sous forme d'algorithme, de façon structurée et indépendante de toute contrainte matérielle ou logicielle.

À l'issue de ce cours, vous serez capable :

- ✗ De comprendre et d'analyser un algorithme préexistant ;
- ✗ D'écrire un algorithme basé sur un code;
- ✗ D'identifier les données d'entrée, de sortie, les traitements attendus, ... ;
- ✗ D'identifier les boucles, les tests, des opérations d'écriture, d'affichage, ... ;
- ✗ De pouvoir valider une solution algorithmique à l'aide de tests simples et intuitives;
- ✗ D'être capable de découper un long algorithme en de sous-algorithmes et de pouvoir les réutiliser plus tard.

ALGORITHMIQUE

GÉNÉRALITÉ > PRÉAMBULE

✖ Informatique

Le terme informatique a été proposé en 1962 par Philippe Dreyfus pour désigner le traitement automatique de l'information: il est construit sur la contraction de l'expression « **information automatique** ». **L'informatique est donc la science du traitement automatique de l'information.**

✖ Matériel Informatique

Le matériel informatique est un ensemble de dispositifs physiques utilisés pour traiter automatiquement des informations.

✖ Logiciel

Le logiciel est un ensemble structuré d'instructions décrivant un traitement d'informations à faire réaliser par un matériel informatique pour traiter automatiquement des informations.

✖ Ordinateur

Un ordinateur est une machine effectuant des opérations simples sur des séquences de signaux électriques conditionnés de manière à ne pouvoir prendre que deux états seulement. Ces séquences de signaux obéissent à une logique binaire du type « **tout ou rien** » et peuvent donc être considérés conventionnellement comme des suites de nombres ne prenant que des 0 et 1.

ALGORITHMIQUE

GÉNÉRALITÉ > PRÉAMBULE

✖ Programmation

Expliquer un ordinateur en détail ce qu'il doit faire, en sachant qu'il ne « comprend » pas le langage humain, mais qu'il peut seulement effectuer un traitement automatique sur des séquences de 0 et de 1.

✖ Programme

Une suite d'instructions, encodées en respectant de manière très stricte un ensemble de conventions fixées à l'avance par un langage informatique. La machine décode alors ces instructions en associant à chaque « mot » du langage informatique une action précise.

✖ Code source

Le texte produit dans un éditeur pour écrire un programme.

✖ Langage de programmation

Un langage de programmation est un langage informatique, permettant à un humain d'écrire un code source qui sera analysé par un ordinateur.

ALGORITHMIQUE

GÉNÉRALITÉ > PRÉAMBULE

✖ Bit

Un bit est un chiffre binaire (0 ou 1). C'est l'unité élémentaire d'information.

✖ Octet

Un octet est une unité d'information composée de 8 bits.

✖ Compilateur

Un compilateur est un programme informatique qui traduit un langage, le langage source, en un autre langage, appelé le langage cible.

✖ Interpréteur

Un interpréteur est un outil informatique (logiciel ou matériel) ayant pour tâche d'analyser et d'exécuter un programme écrit dans un langage source.

ALGORITHMIQUE

GÉNÉRALITÉ > NOTION D'ALGORITHME

ALGORITHME

Un algorithme est une suite ordonnée d'instructions qui indique la démarche à suivre pour résoudre une série de problèmes équivalents.

- + Un algorithme n'est pas forcément destiné à décrire la solution d'un problème pour la programmation et l'informatique.
 - × Un algorithme en cuisine s'appelle une recette
 - × Un algorithme en musique s'appelle une partition
 - × Etc.

ALGORITHMIQUE

GÉNÉRALITÉ > NOTION D'ALGORITHME

HISTORIQUE

- ✗ Les premières formulations de règles précises pour la résolution de certains d'équations remontent aux *Babyloniens* (1800 avant J.C.).
- ✗ *Archimède* qui proposa un algorithme pour le calcul de π^2 .
- ✗ *Euclide* proposa un algorithme pour trouver le PGCD de deux entiers relatifs.
- ✗ Le mot algorithme vient du nom d'un mathématicien arabophone du IX^e siècle *Al-Khuwarizmi*. Il est le premier à avoir systématisé des algorithmes dans son ouvrage « **Abrégé du calcul par la restauration et la comparaison** », il étudie toutes les équations du second degré et en donne la résolution par des algorithmes généraux.

ALGORITHMIQUE

GÉNÉRALITÉ > NOTION D'ALGORITHME

ALGORITHMIQUE

L'algorithmique est la science des algorithmes. Elle s'intéresse à l'art de construire des algorithmes ainsi qu'à caractériser leur :

- + Terminaison
- + Validité
- + Robustesse
- + Réutilisabilité
- + Complexité
- + Efficacité
- + Etc.

ALGORITHMIQUE

GÉNÉRALITÉ > NOTION D'ALGORITHME

ALGORITHMIQUE: TERMINAISON

Un algorithme doit fournir un résultat au bout d'un nombre finit d'instructions. Il ne doit pas exister une instance pour laquelle l'exécution rentre dans une boucle infinie.

+ Exemple:

Considérons un algorithme qui permet d'aller à la gare routière. Il faut que l'on y soit au bout d'un certain nombre de pas. Par contre, Si nous faisons le tour de la ville jusqu'à revenir à notre point de départ sans jamais se rendre à la gare routière, c'est que notre algorithme ne se termine jamais.

ALGORITHMIQUE

GÉNÉRALITÉ > NOTION D'ALGORITHME

ALGORITHMIQUE: VALIDITE

La validité d'un algorithme est son aptitude à réaliser exactement la tâche pour laquelle il a été conçu.

+ Exemple:

Considérons à nouveau notre algorithme qui permet d'aller à la gare routière. En exécutant scrupuleusement les instructions on devrait arriver exactement à la gare routier et non au marché.

ALGORITHMIQUE

GÉNÉRALITÉ > NOTION D'ALGORITHME

ALGORITHMIQUE: ROBUSTESSE

La robustesse d'un algorithme est son aptitude à se protéger des conditions anormales d'utilisation.

+ Exemple:

Considérons un algorithme qui fabrique de la boisson à partir de mangues. Si, à la place des mangues nous fournissons des oranges, comment va-t-il se comporter ?

ALGORITHMIQUE

GÉNÉRALITÉ > NOTION D'ALGORITHME

ALGORITHMIQUE: REUTILISABILITE

La réutilisabilité d'un algorithme est son aptitude à être réutilisé pour résoudre des tâches équivalentes à celle pour laquelle il a été conçu.

+ Exemple:

Un algorithme qui permet de calculer la factorielle d'un entier naturel devrait pouvoir le faire pour n'importe quel entier.

ALGORITHMIQUE

GÉNÉRALITÉ > NOTION D'ALGORITHME

ALGORITHMIQUE: COMPLEXITE

La complexité d'un algorithme est le nombre d'instructions élémentaires à exécuter pour réaliser la tâche pour laquelle il a été conçu.

+ Exemple:

Si un « élève égaré » est un piéton, la complexité de l'algorithme de recherche de chemin peut se compter en nombre de pas pour arriver à l'endroit de destination.

ALGORITHMIQUE

GÉNÉRALITÉ > NOTION D'ALGORITHME

ALGORITHMIQUE: EFFICACITE

L'efficacité d'un algorithme est son aptitude à utiliser de manière optimale, les ressources du matériel qui l'exécute. Elle est mesurée par sa durée de calcul, sa consommation de mémoire RAM, ...

+ Exemple:

Combien de variables et d'opérations pour calculer:

$$1 + 2 + 3 + \dots + n ?$$

Solution non optimale : 2 variable, $n-1$ opérations

Solution optimale : 1 variable, 3 opérations

ALGORITHMIQUE

GÉNÉRALITÉ > NOTION D'ALGORITHME

ALGORITHMIQUE: AFFINITE

Un algorithme doit décrire précisément le traitement qu'il doit exécuter et s'assurer que **tous les cas de figures possible** ont bien été prévus.

+ Exemple:

Algorithme permettant de calculer la durée d'un voyage à partir du tableau d'affichage.

1. Consulter l'heure de départ
2. Consulter l'heure d'arrivé
3. Soustraire l'heure de départ de celle d'arrivée

ALGORITHMIQUE

GÉNÉRALITÉ > NOTION D'ALGORITHME

METHODOLOGIE

Une *démarche algorithmique* suit un certain nombre d'étapes que sont :

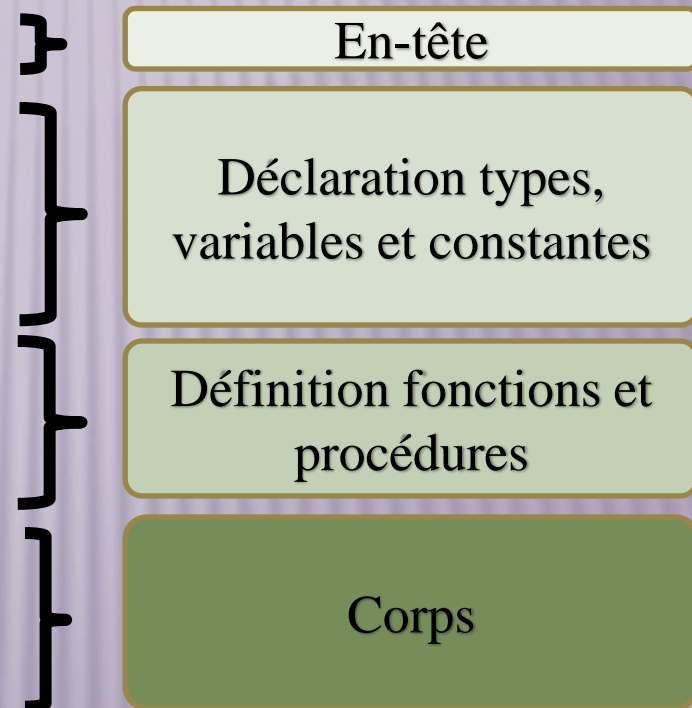
- ✖ Identifier les données d'entrée
- ✖ Identifier les données de sortie
- ✖ Déterminer les actions ou opérations élémentaires
- ✖ Spécifier l'enchaînement de ces actions

ALGORITHMIQUE

GÉNÉRALITÉ > NOTION D'ALGORITHME

STRUCTURE

Algorithme nom de l'algorithme
[**Constante** définition constantes]
[**Types** définition de types]
[**Variable** définition variables]
[**Fonction** définition fonctions]
[**Procédure** définition procédures]
Début
 {instructions}
Fin



ALGORITHMIQUE

GÉNÉRALITÉ > NOTION D'ALGORITHME

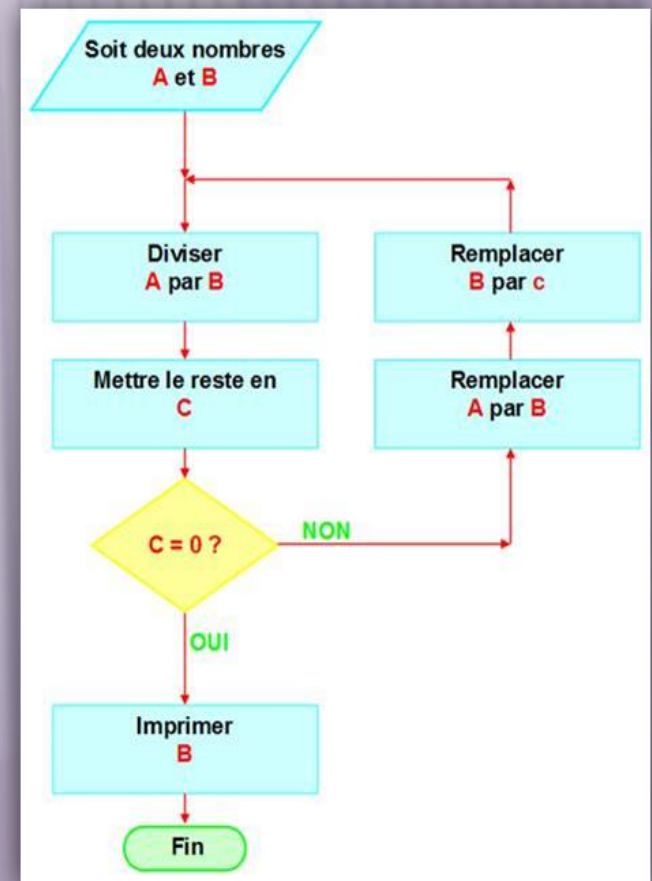
QUELLE CONVENTION D'ECRITURE ?

Historiquement, plusieurs types de notations notamment une représentation graphique qu'on appelait **organigramme** ou **logigramme**.

Aujourd'hui, cette représentation est quasiment abandonnée car non adéquate lorsque l'algorithme grossit.

On utilise généralement une série de conventions appelée «pseudo-code», qui ressemble à un langage de programmation et qui est susceptible de varier légèrement d'un livre à un autre.

C'est bien normal car le «pseudo-code», est purement conventionnel; aucune machine n'est censée le reconnaître.



ALGORITHMIQUE

BASES > TYPES ET VARIABLES

TYPE

- ✗ Dans un programme informatique, on va avoir en permanence besoin de stocker provisoirement de l'information.
- ✗ Il peut s'agir de données issues du disque dur, ou fournies par l'utilisateur, ou résultantes d'un calcul, etc.
- ✗ Ces données peuvent être de plusieurs types: nombre, texte, etc. .
- ✗ Mais c'est quoi un type alors?

ALGORITHMIQUE

BASES > TYPES ET VARIABLES

TYPE: DEFINITION

Un type est un nom pour un ensemble de valeurs. Il sert à préciser :

- ❖ La nature des valeurs acceptables
- ❖ Les opérations autorisées sur ces valeurs
- ❖ La taille mémoire utilisée

Il peut être :

- ❖ **Simple** : entier, réel, booléen , caractère, chaîne de caractères
- ❖ **Composé**: tableau, énumération, etc.

ALGORITHMIQUE

BASES > TYPES ET VARIABLES

VARIABLE

- ❖ Les données qu'utilise un programme peuvent varier au cours de son exécution. On les appelle alors des variables.
- ❖ Une variable est le nom d'un «réceptacle» destiné à contenir une valeur. Lorsque nous nous intéresserons un peu plus à l'ordinateur, le réceptacle sera une zone mémoire.
- ❖ Elle se déclare de la manière suivante :

Variable nom Variable : type

Exemple:

- ❖ Variable age : entier
 - ❖ Variable moyenne, ecartType : réel
 - ❖ Variable operateur : caractère
 - ❖ Variable sms : chaîne de caractères
 - ❖ Variable estDemarre : booléen
- ❖ **Convention:** comme dans la plus part des langage de programmation, on ne peut pas appeler une variable n'importe comment :
 - ❖ Le nom de variable peut contenir des minuscules, des majuscules et des chiffres ;
 - ❖ Il doit commencer par une lettre ;
 - ❖ S'il doit comporter plusieurs mots, on n'utilise jamais d'espaces entre les mots. On peut utiliser soit le caractère "underscore : _ « ou coller les mots en mettant en majuscule le premier caractère;
 - ❖ Les accents ("é", "à", "ê", etc) et les symboles (+, -, *, /, %, =) sont aussi interdits.

ALGORITHMIQUE

BASES > TYPES ET VARIABLES

AFFECTATION

- ❖ L'affectation est une instruction qui permet de changer la valeur d'une variable.
- ❖ Elle modifie le contenu du récipient désigné par la variable.
- ❖ Elle est une instruction dite «destructrice».

Variable ← valeur

Exemple:

- ❖ Variable note : réel
- ❖ note ← 17.75

ALGORITHMIQUE

BASES > TYPES ET VARIABLES

LES CONSTANTES

- ❖ Une constante est une variable particulière dont le contenu n'est pas modifiable. Elle se déclare comme suit :

Constante **NOM_CONSTANTE = valeur**

Exemple:

- ❖ Constante $\text{PI}=3.1415$
- ✖ Convention : une constante est écrite en majuscule. Lorsque son nom comporte plusieurs noms, on les sépare par " _".
- ✖ Remarque : En algorithmique, le type d'une constante est optionnel, on interprète directement la valeur assignée.

ALGORITHMIQUE

BASES > LECTURE-ECRITURE

INSTRUCTION D'ECRITURE

- ✗ Les informations fournies à l'utilisateur peuvent être de natures diverses: texte brut, contenu d'une variable ou résultat d'un calcul.
- ✗ Ceci est réalisé grâce à la fonction d'écriture ou d'affichage dont voici une syntaxe:

écrire(variable)

écrire("Bonjour tout le monde")

écrire("Vous avez ", age, " ans")

ALGORITHMIQUE

BASES > LECTURE-ECRITURE

INSTRUCTION DE LECTURE

- ✘ Un programme est censé recevoir des données provenant de l'utilisateur.
- ✘ Un besoin de stocker ces dernières peut se faire sentir et dans ce cas, il faut nécessairement les enregistrer dans une variable.
- ✘ Pour ce faire, nous utilisons une fonction dont voici une syntaxe:

lire(variable)

lire(variable1,variable2,...)

ALGORITHMIQUE

CAS PRATIQUES N°1

Application 1 :

Quelles seront les valeurs des variables A et B après exécution des instructions suivantes ?

Algorithme Appli1

Variables A, B : Entier

Début

A ← 15

B ← A + 8

A ← 12

Fin

ALGORITHMIQUE

CAS PRATIQUES N°1

Application 2 :

Quelles seront les valeurs des variables A, B et C après exécution des instructions suivantes ?

Algorithme Appli2

Variables A, B, C : Entier

Début

A ← 10

B ← 9

C ← A + B

A ← 21

C ← B - A

Fin

ALGORITHMIQUE

CAS PRATIQUES N°1

Application 3 :

Quelles seront les valeurs des variables A et B après exécution des instructions suivantes ?

Algorithme Appl3

Variables A, B : Entier

Début

A ← 10

B ← 20

A ← B

B ← A

Fin

1. Les deux dernières instructions permettent-elles d'échanger les deux valeurs de B et A?
2. Si l'on inverse les deux dernières instructions, cela change-t-il quelque chose ?

ALGORITHMIQUE

BASES > OPERATEURS-EXPRESSIONS

OPERATEURS ARITHMETIQUES

- ✖ Addition : +
- ✖ Soustraction : -
- ✖ Multiplication : *
- ✖ Division entière : div
- ✖ Division réelle : /
- ✖ Modulo : mod ou %
- ✖ Puissance : ^
- ✖ Racine carrée : sqrt

ALGORITHMIQUE

BASES > OPERATEURS-EXPRESSIONS

OPERATEURS DE COMPARAISON

- ✖ Strictement supérieur : $>$
- ✖ Strictement inférieur : $<$
- ✖ Supérieur ou égal : \geq
- ✖ Inférieur ou égale : \leq
- ✖ Différent : \neq
- ✖ Egal : $=$

ALGORITHMIQUE

BASES > OPERATEURS-EXPRESSIONS

OPERATEURS: TYPES

Il existe deux types d'opérateurs:

- ❑ *Unaire* : il n'agit que sur une donnée;
 - ❖ Exemple : la négation(NON)
- ❑ *Binaire* : il agit sur deux données;
 - ❖ Exemple : addition, division, etc.

ALGORITHMIQUE

BASES > CONCATÉNATION

CONCATENATION

Le terme **concaténation** désigne l'action de mettre des informations les une à la suite des autres. En algorithmique, l'opérateur utilisé est **&**

❑ Exemple

Variable s1,s2,s :chaîne de caractères

$s1 \leftarrow \text{"Bonjour"}$

$s2 \leftarrow \text{"tout le monde"}$

$s \leftarrow s1 \& s2$

s vaut maintenant "Bonjour tout le monde"

ALGORITHMIQUE

BASES > COMMENTAIRE

COMMENTAIRE

Un commentaire est un simple texte qui sert à donner une information. Ci-dessous, sa syntaxe.

```
{commentaire}
```

❑ Exemple

Algorithme PGCD

{Cet algorithme permet de calculer le PGCD de deux entiers}

Début

{instructions}

Fin

ALGORITHMIQUE

BASES > CONDITION

CONDITION

Une condition est une comparaison. Elle est composée de trois éléments :

- ❖ Une valeur
- ❖ Un opérateur de comparaison
- ❖ Une autre valeur

Les valeurs peuvent être à priori de n'importe quel type. Mais si l'on veut que la comparaison ait un sens, il faut que les deux valeurs soient du même type.

ALGORITHMIQUE

BASES > CONDITION ET EXPRESSION BOOLEENNE

CONDITION ET EXPRESSION BOOLEENNE

- Un condition est une expression booléenne pouvant prendre VRAI ou FAUX.
- Elle peut être constituée d'une seule comparaison ou de plusieurs.
- Dans le dernier, nous parlons d'expression conditionnelle.

Exemple:

Variable a, b: booléen

$a \leftarrow 1 > 2$

$b \leftarrow (1 > 2) \text{ ET } (3 = 7) \text{ OU } ((3 = 5) \text{ ET } (3 < 5))$

ALGORITHMIQUE

CAS PRATIQUES N°2

Application 4 :

Trouver les valeur booléennes prises au cours de l'algo suivant :

Algorithme calcul-valeurs-boolennes

Variable

a, b : Entier

b1, b2, b3, b4 : Booléen

Début

a ← 10

b ← 4

b1 ← (10>10) ET (5=5)

b2 ← (a=10) OU (b=5) ou (3=5)

b3 ← (a>b) ET (5=5) ou (b<a)

b4 ← (FAUX) ET (VRAI) ou (a>b)

Fin

ALGORITHMIQUE

CAS PRATIQUES N°2

✖ Application 5 :

Ecrivez un algorithme qui à partir de la valeur saisie du côté d'un carré donné, calcule son périmètre et sa surface et affiche les résultats à l'écran.

✖ Application 6:

Ecrivez un algorithme qui à partir de la valeur de l'arrêt d'un cube saisie au clavier, calcule sa surface de base et son volume et affiche les résultats à l'écran.

✖ Application 7:

Ecrivez un algorithme qui à partir de la valeur du rayon d'un cercle saisie au clavier, calcule son diamètre, sa surface et sa circonférence.

ALGORITHMIQUE

CAS PRATIQUES N°2

✖ Application 8 :

Ecrivez un algorithme qui à partir du prix hors taxe PHT d'un produit et du taux de TVA calcule et affiche le prix toute taxe comprise PTTC.

✖ Application 9:

Ecrivez un algorithme permettant de déclarer trois variables A, B, C de type réel, d'initialiser leurs valeurs et ensuite d'effectuer la permutation circulaire des trois variables.

✖ Application 10:

Ecrire l'algorithme permettant de saisir l'abscisse d'un point A et de calculer son ordonné $f(x) = 2X^3 - 3X^2 + 4$. Evaluer le résultat en expliquant les ordres de priorité pour $x = -2$.

Affaires à suivre



Feedback sur:
pape.abdoulaye.barro@gmail.com