

Cours: Algorithmique (Partie 1)

EL OUKKAL Sanae,
s.eloukkal@gmail.com

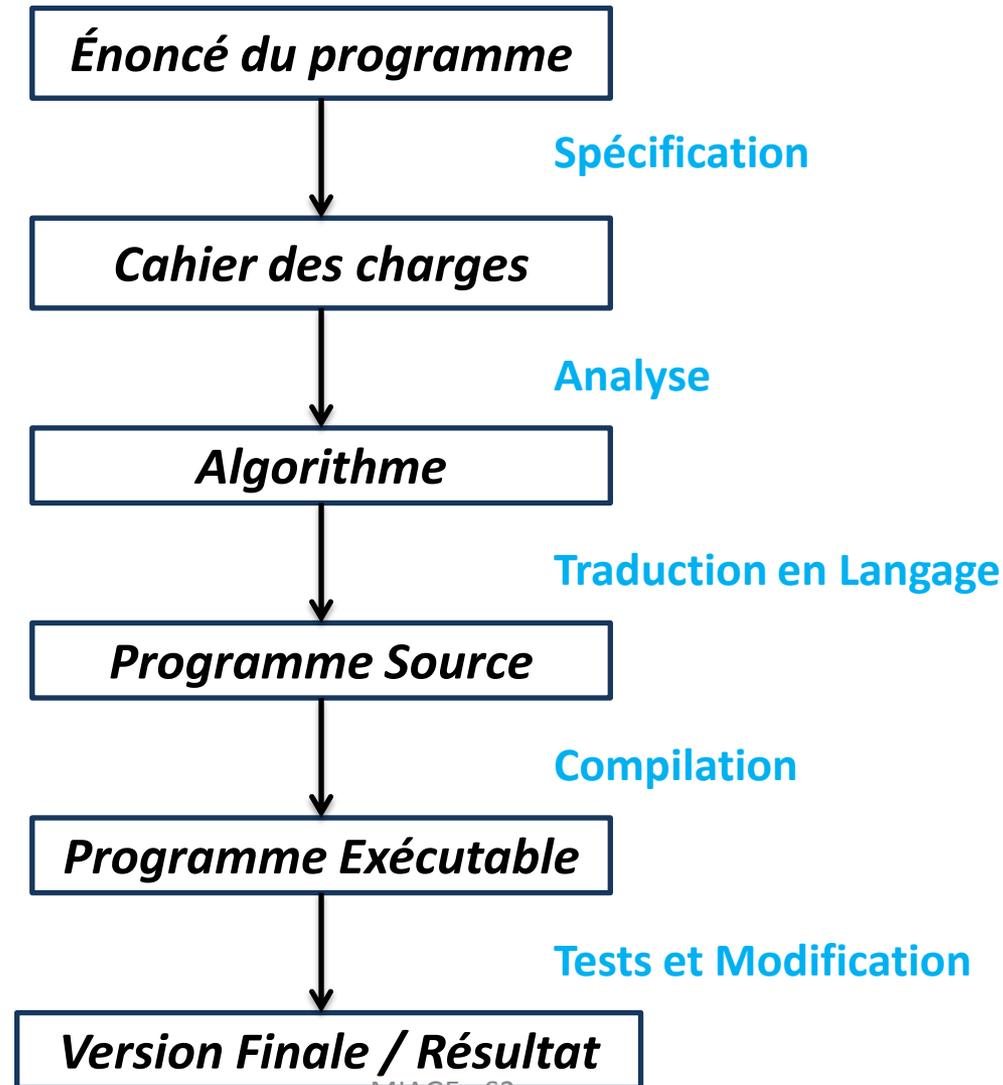
Plan du Cours

- Les notions de bases
- Les outils de bases de l'algorithmique
- Les instructions élémentaires en algorithmique
- Les instructions conditionnelles et alternatives
- Fonctions et Procédures
- Tableaux
- ...

Objectif du Cours

- Savoir analyser les problèmes
 - Suivre une logique structurée pour la résolution de problèmes
 - Savoir présenter une solution avec un raisonnement explicite
- ➔ concevoir et écrire des algorithmes

Étapes pour résoudre un problème (écrire un programme)



L'algorithme de tous les jours!!

- Indiquer un chemin à quelqu'un
- Faire chercher un objet à quelqu'un par téléphone
- Recette de cuisine
- Mode d'emploi d'un appareil électronique
- Notice de montage d'un meuble en kit
- ...

Algorithme

- Un algorithme est une description complète et détaillée des actions à effectuer et de leur séquençement pour arriver à un résultat donné
- Processus décrivant étape par étape comment résoudre un problème

Algorithme

- Pour fonctionner, un algorithme doit contenir uniquement des instructions compréhensibles par celui qui devra l'exécuter .
- Il doit être très clair pour certains voir tout le monde.

Algorithme

- L'ordinateur est rapide, mais n'est pas intelligent, Il faut lui dire quoi faire, et comment le faire.
- Il faut lui fournir un algorithme : décrire étape par étape comment résoudre le problème
 - Si l'algorithme est juste, le résultat est le résultat voulu,
 - Si l'algorithme est faux, le résultat est aléatoire

Algorithme

Maitriser l'algorithme

Intuition

**Méthodique et
rigoureux**

Représentation d'algorithme

- Deux façons pour représenter un algorithme:
 - **L'Organigramme: représentation graphique avec des symboles (carrés, losanges, etc.)**
 - Offre une vue d'ensemble de l'algorithme
 - Représentation quasiment abandonnée aujourd'hui
 - **Le pseudo-code: représentation textuelle avec une série de conventions ressemblant à un langage de programmation (sans les problèmes de syntaxe)**
 - Plus pratique pour écrire un algorithme / traduire en un langage
 - Représentation largement utilisée

En Résumé

- Construire un algorithme dépend des étapes suivantes :

**Données
(Input)**



**Opérations
(Instructions Finies)**



**Résultats
(Output)**

- Contient uniquement des instructions compréhensibles par celui qui devra l'exécuter.

Algorithmique et Programmation

- Pourquoi apprendre l'algorithmique pour apprendre à programmer ?
 - Parce que l'algorithmique exprime les instructions résolvant un problème donné indépendamment des particularités de tel ou tel langage.
 - Apprendre l'algorithmique, c'est apprendre à manier la structure logique d'un programme informatique.
 - L'algorithmique utilise un ensemble de mots clés et de structures permettant de décrire de manière complète, claire, l'ensemble des opérations à exécuter sur des données pour obtenir des résultats.

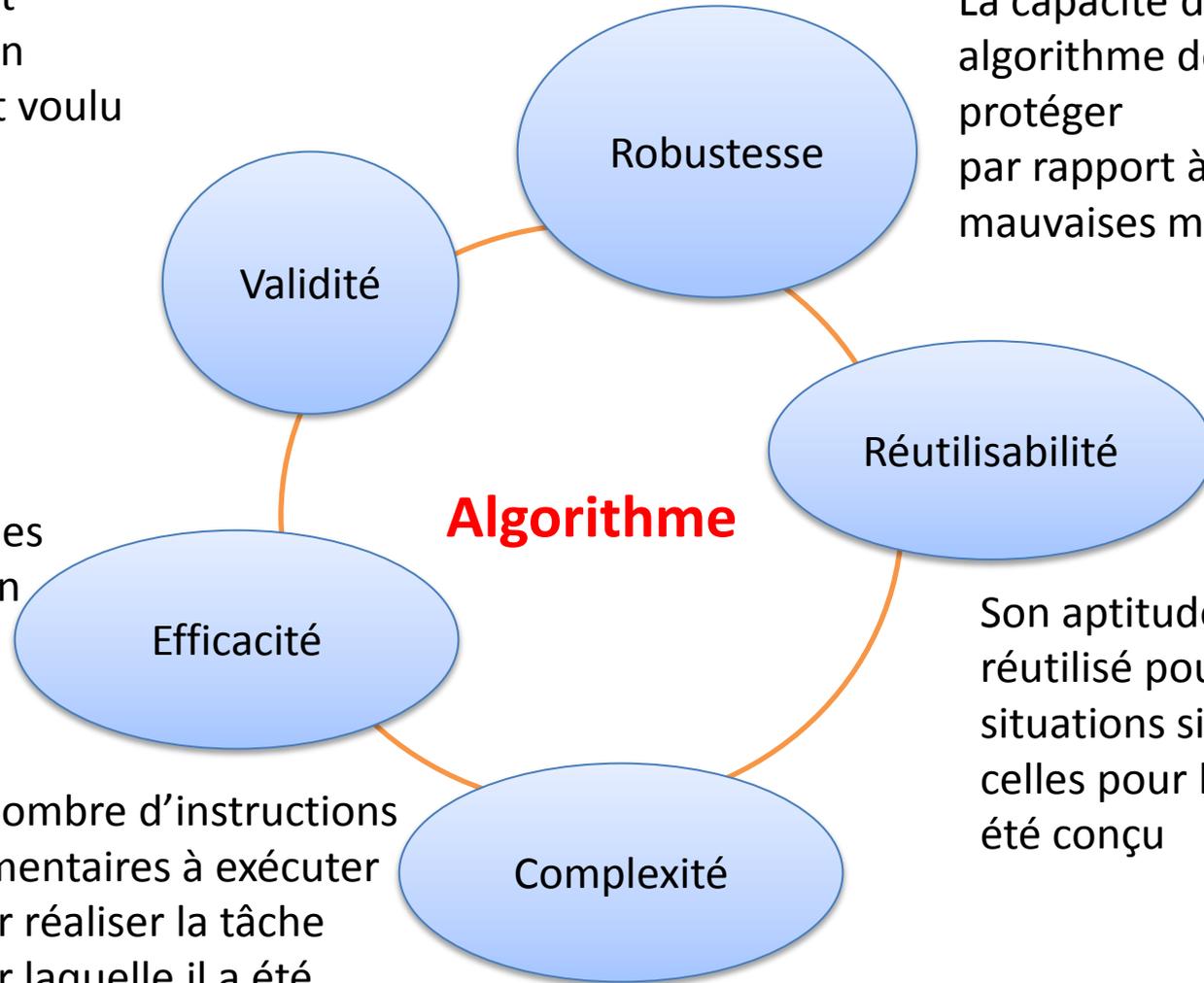
Caractéristiques d'un Algorithme

- Validité
- Robustesse
- Réutilisabilité
- Complexité
- Efficacité

Caractéristique d'un Algorithme

Un algorithme n'est valide que lorsqu'on retrouve le résultat voulu pendant son exécution

La capacité d'un algorithme de se protéger par rapport à de mauvaises manipulations



L'optimisation des ressources matérielles pour l'exécution d'un algorithme.

Le nombre d'instructions élémentaires à exécuter pour réaliser la tâche pour laquelle il a été conçu.

Son aptitude à être réutilisé pour d'autres situations similaires à celles pour lesquelles il a été conçu

NOTIONS DE BASE

Notion de Variable

- Un Emplacement pour stocker des valeurs ou données, des valeurs intermédiaires ou des résultats.
- Une variable correspond toujours à une valeur
- Son contenu peut être modifié par une action durant l'exécution d'un algorithme.

Notion de Variable

- La valeur de la variable appartient à un domaine particulier défini par un *type de base*.
- les quatre types de base de toute algorithmique informatique sont:
 - Les booléens (BOOLEEN)
 - Les entiers (ENTIER)
 - Les flottants (REEL)
 - Les caractères (CARACTERE)

Type des Variables

- Type numérique (entier ou réel)
 - Nombre entier:
 - Byte(codé sur 1 octet): de 0 à 255
 - Entier court(codé sur 2 octets) : -32 768 à 32 767
 - Entier long (codé sur 4 ou 8 octets)
 - Nombre à virgule:
 - Réel simple précision(codé sur 4 octets)
 - Réel double précision(codé sur 8 octets)
- Type logique ou booléen:
 - Deux valeurs : **VRAI** ou **FAUX**

Type des Variables

- Type caractère:
 - Lettres majuscules, minuscules, chiffres, symboles, ...

Exemples: 'A', 'a', '1', '?'

- Type chaîne de caractère:
 - Toute suite de caractères,

Exemples: "MIAGE", "Casablanca"

Type des Variables

- Type Date:
 - **Jour/Mois/Année**
- Type Monétaire:
 - **Numérique**

Déclaration d'une Variable

- Les variables doivent être déclarées avant d'être utilisées,
- Elles doivent être caractérisées par :
 - un nom (**Identificateur**)
 - un type (**entier, réel, caractère, chaîne de caractères, ...**)

Variables liste d'identificateur : type;

Déclaration d'une Variable

- Les noms des variables peuvent comporter des lettres et des chiffres,
- Un nom doit commencer par une lettre alphabétique ,

Valide: A1

Invalide: 11S

Déclaration d'une Variable

- Doit être constitué uniquement de lettres, de chiffres et du soulignement _ (Eviter les caractères de ponctuation et les espaces),

Valide:
A_2018

Invalide:
A 2018

- Doit être différente des mots réservés aux langages de programmation.

Déclaration d'une Variable

- **Conseil:**

- Pour la lisibilité du code, choisir des noms significatifs qui décrivent les données manipulées
- Toute variable utilisée dans un algorithme/programme doit avoir fait l'objet d'une déclaration préalable

Déclaration d'une Variable

Variables liste d'identificateur : type

- Exemple:

Variables i, j, k : entier;

x, y : réel;

OK : booléen;

ch1, ch2 : chaîne de caractères;

CONSTANTE

Définition d'une Constante

- C'est une donnée fixe dont la valeur ne varie pas durant l'exécution d'un algorithme.
- Une constante est caractérisée par son nom et sa valeur fixe.

Déclaration de la Constante

Constante (**Nom_Constante** : type) ← valeur;
Constante **Nom_Constante** = valeur ;

- Exemple :
 - Constante Pi = 3.14 ;
 - Constante Mois = "Avril" ;

Remarque

- " 150" est différent de 150
- 1 est différent de '1'
- **Conseil:**
 - Pour éviter de confondre un nombre d'une chaîne, il faut **TOUJOURS** noter une chaîne entre guillemets!

L'INSTRUCTION D'AFFECTATION

L' Affectation

- L'affectation consiste à attribuer une valeur à une variable
 - ça consiste à remplir ou à modifier le contenu d'une zone mémoire
- L'affectation se note avec le signe ←
Variable ← exp : attribuer la valeur de exp à la Variable

L' Affectation

Variable ← *exp* ;

- *exp* peut être une valeur, une autre variable ou une expression
- Variable et *exp* doivent être du même type ou de types compatibles
- L'affectation ne modifie que ce qui est à gauche de la flèche

Remarque

- L'affectation n'est pas commutative:
 - $A \leftarrow B$ est différente de $B \leftarrow A$
 - L'affectation est différente d'une équation mathématique
 - $A = A+1$: possible
 - $A+1 = 2$: pas possible

Affectation : Exemple

Variables i, j, k : entier
x, y : réel
OK : booléen
ch1, ch2 : chaîne de caractères

- $k \leftarrow i+j;$
- $OK \leftarrow \text{“MIAGE”};$
- $x \leftarrow 10.3;$
- $ch2 \leftarrow ch1 ;$
- $x \leftarrow 4;$
- $x \leftarrow j;$
- $i \leftarrow 10.3 ;$
- $OK \leftarrow \text{FAUX} ;$
- $ch1 \leftarrow \text{“MIAGE”};$
- $j \leftarrow x;$
- $i \leftarrow 1;$
- $j \leftarrow i;$

Ordre des Instructions

- L'ordre dans lequel les instructions sont écrites va jouer un rôle essentiel dans le résultat final.

Variable A : Entier;

Début

A ← 28;

A ← 7;

Fin

Variable A : Entier;

Début

A ← 7;

A ← 28;

Fin

Ordre des Instructions :

Exemple

Exemple 1

- $A \leftarrow 3;$
- $B \leftarrow 5;$
- $A \leftarrow B;$
- $B \leftarrow A;$

Exemple 3

- $A \leftarrow 1;$
- $B \leftarrow 3+A;$
- $A \leftarrow 3;$

Exemple 2

- $A \leftarrow 3;$
- $B \leftarrow A;$
- $A \leftarrow A+B;$
- $B \leftarrow A+B;$

Exemple 4

- $A \leftarrow 5;$
- $B \leftarrow 3;$
- $C \leftarrow A+B;$
- $A \leftarrow 2;$
- $B \leftarrow 2;$
- $C \leftarrow B-1;$

Ordre des Instructions:

Exercice 1

- Donnez les valeurs des variables A, B et C après exécution des instructions suivantes?

Variables A, B, C : Entier;

Début

A \leftarrow 3;

B \leftarrow 7;

A \leftarrow B;

B \leftarrow A+5;

C \leftarrow A + B;

C \leftarrow B - A;

Fin

Ordre des Instructions:

Exercice 2

- Donnez les valeurs des variables A et B après exécution des instructions suivantes? Échange?

Variables A, B :Entier;

Début

A ← 1;

B ← 2;

A ← B;

B ← A;

Fin

Ordre des Instructions:

Exercice 3

- Ecrire un algorithme qui permet de permuter les valeurs de deux variables A et B

LES INSTRUCTIONS DE LECTURE ET D'ÉCRITURE

Les Instructions d'E/S

- Les instructions de lecture et d'écriture permettent à la machine de communiquer avec l'utilisateur
- Dès que le programme rencontre une instruction Lire, l'exécution s'interrompt, attendant la frappe d'une valeur au clavier
- Syntaxe:
 - Lire (variable1) ;
 - Lire (variable2) ;

Les Instructions d'E/S

- L'écriture permet d'afficher des résultats à l'écran afin de les communiquer à l'utilisateur.
- Syntaxe:
 - Ecrire (variable) ;
 - Ecrire ("message") ;
 - Ecrire ("message", variable) ;
- Conseil:
 - Avant de lire une variable, il est fortement conseillé d'écrire des messages à l'écran, afin de prévenir l'utilisateur de ce qu'il doit frapper



Université
Internationale
de Casablanca

UNIVERSITE INTERNATIONALE DE CASABLANCA

À suivre...