# La codification

**Objectif :**

 La structuration de l’information en vue de son traitement automatique et sa définition de manière claire et précise, pour cela, on associe aux différents objets ou concepts rencontrés dans une application des codes permettant de les caractérisés d’une manière unique

**Définition :**

Un code est une représentation abrégée d’une information.

 On peut distinguer deux applications principales de la codification :

**1er:**le besoin de nommer de manière unique une variable qu’on manipule dans un algorithme ou dans un programme.

**Exemple :** manipuler le numéro de l’étudiant : Num Etud

**2éme :** c’est le besoin de définir les règles que doit vérifier la valeur d’une donnée manipulée.

**Exemple :** Immatriculation d’une voiture

3 chiffres 1ch 2chi 2chiffres

**Remarque :**

La codification d’un attribut peut concerner son nom ou/et sa valeur.

**Caractéristique d’une codification :**

1. **Non ambiguïté :** une codification est non ambiguë si elle associe à chaque objet un code et un seul, et à chaque code un objet et un seul (Bijection entre l’ensemble des codes et l’ensemble des objets)
2. **Adaptation aux besoins des utilisateurs :** cela nécessite :
	* Simplicité de la fonction de codification

Exemple : Num Etud → numéro d’étudiant

Le cas le plus simple : fonction de correspondance (explicite)

* + Facilité d’interprétation des codes (pour retrouver les objets) fonction de décodification facile à déterminer.
1. **Possibilité d’extension et d’insertion :**
	* Extension : l’ensemble des objets codifies peut s’accroître sans remettre en cause la codification choisie.
	* Insertion : les objets nouveaux peuvent s’insérer entre les objets existants sans remettre en cause la codification choisie

1. **La concision :** l’intérêt d’une codification est d’éviter d’avoir à manipuler des informations trop langue, pour être efficace un code doit donc comporter le moins de caractères possibles.

 **Remarque :** la longueur d’un code doit toujours être défini en fonction des possibilités d’évolution de l’ensemble des objets codifiés

1. **Mnémonique :**

S’associe à mémoire, la lecture du code doit faire penser (rappeler) à l’objet codifié, ceci est vrai pour : NumEtud qui codifie numéro Etudiant,

**Remarque :**

1. Le caractère mnémonique s’applique généralement à la codification des noms d’attributs et non au valeurs.
2. La codification est avant tout, une convention d’écriture entre la personne chargée du codage de l’information et celle chargée de son décodage.

L’origine des règles de codage importe donc pour seul compte c’est l’efficacité.

**Différents types de codification :**

1. **Codification séquentielle :**

Elle consiste à affecter des numéros consécutifs aux objets à codifier.

**Exemple :** clé : (0101,0102,..., etc.)

**Avantages :**

* + Simplicité
	+ Non-ambiguïté
	+ Possibilité d’extension

**Inconvénients :**

* + Non significatif
	+ Impossibilité d’insertion
	+ Pas de regroupements possibles
1. **Codification par tranches :**

Elle consiste à réserver des tranches de code à des catégories d’objets à l’intérieur d’une même tranche, la codification est généralement séquentielle.

**Exemple :** Gestion des stocks d’une pharmacie :

- No 0001→0999 : Médicament sans ordonnance

0001→0099 : Antalgique

0100→ 0599 : Anti-inflammatoire

0600→ 0799 : Fortifiant.

………..

* + No 1100→1500 : Médicament interdits sans ordonnance

**Remarque :** les tranches peuvent être décomposées ou non

**Avantage :**

* + Non ambiguïté
	+ Simplicité
	+ Possibilité d’extension et d’insertion.

**Inconvénients :** - non significatif sans table de correspondance.

1. **codification articulée :**

**Remarque :** c’est la codification la plus utilisée en informatique

**Définition :** chaque code est découpé en zones appelé descripteur, et chaque descripteur possède un sens particulier.

**Exemple :**

* + 1. immatriculation d’une voiture

no séquentiel type année no wilaya

* + 1. code de l’employé

A n n n

Num seq

Catégorie

Catégorie

D : direction

O : ouvrier

A : administratif

**Avantages :**

* + Non ambiguë
	+ Possibilité d’insertion et d’extension
	+ Très répondue
	+ Significative à condition d’un choix efficace des descriptions
	+ Possibilité de regroupement et de contrôle.

**Inconvénients :**

* + Code longet lourd à manipulé
	+ Risque de saturation
	+ L’instabilité : le changement d’une caractéristique de l’objet peut remettre en question toute la codification.
1. **Codification à niveau :**

C’est un cas particulier de codification articulée, les descripteurs sont des niveaux hiérarchiques

**Exemple :** contenu d’un livre

Chapitre section paragraphe sous parag titre

**Avantages :**

* + Même avantage que la codification articulée
	+ Facilité de recherche arborescente

**Inconvénients :**

* + Même inconvénients que la codification articulée.
1. **Codification mnémonique :**

Elle consiste à représenter le nom d’un objet par un petit nombre de caractère que rappelle cet objet.

**Exemple :**

Numéro étudiant→NumEtud

Numéro poste de travail→NumPastra

**Avantages :**

Significative et très pratique (facilité de décodage)

**Inconvénients :**

Porte sur les noms des attributs et non pas sur leurs valeurs.

**Remarques :** la codification mnémonique est très utilisée pour désigner les variables dans des algorithmes ou des programmes.

**Critères de choix d’une codification :**

* + utilisation future du code
	+ Nombre d’objets à codifier et évolution de ce nombre
	+ Statistique
	+ Codification déjà existante
	+ Avis des utilisateurs
	+ Résultat des tests avant adoption.

# Les contrôles

**Rappel :**

 Le but du système d’information est d’associé la délivrance des informations aux utilisateurs qui sont correct, garantir la correction des informations nécessite de leur faire subir des contrôles.

**Définition :**

**Contrôle :** vérification que les caractéristiques d’une information sont conformes à ce qu’elles doivent être.

**Contrôle d’un SI** **=** contrôle des données **+** contrôle des traitements **+** contrôle de résultats.

Le contrôle des traitements étant lié à la justesse d’algorithmes, on s’intéresse ici à la correction des informations traitées et produites.

**Remarque :**

Les contrôles peuvent être manuels (usuel), automatiques (programmé)

**Principaux types de contrôles :**

On distingue deux types de contrôles

1. Les contrôles se rapportant à l’information elle-même et ces caractères indépendamment de toute autre information. (Contrôle direct)
2. Les contrôles déterminant la justesse d’une information par comparaison à d’autres informations (contrôle indirect)

**Contrôle à afficher sur une information :**

1. **Les contrôles directs :**
	1. **Contrôle de présence :**

Il consiste à vérifier que l’information figure bien ( est présente) sur le support dans lequel elle doit apparaître.

**Exemple :** fichier étudiant : si nom n’existe pas  erreur

**Remarque :** contrôle de présence conditionnelle

**Exemple :**

Si nom femme = «   »  erreur

Si fille ≠ M  nom fille qui figure

* 1. **Contrôle de type :**

Il consiste à vérifier le type numérique ou alphabétique d’une information en fonction de son sens.

**Exemple :**

Notel = 2AB356  erreur

Notel = 03503582395  erreur

Car le contrôle ne signale pas d’erreur, on peut par exemple ajouter le contrôle de la langueur No chiffre = 9

* 1. **Contrôle de cadrage :**

Consiste à vérifier la position d’une information dans sa zone de saisie.

Par convention : Numérique ⇒ cadrage à droite 1 3

 Alphabétique ⇒ cadrage à gauche A B

**Remarque :** ce contrôle est généralement assuré de manière automatique par le logiciel.

* 1. **Contrôle de vraisemblance :**

Consiste à s’assurer que la valeur de l’information est très semblable (possible, réalisable) **en fonction de son sens**.

**Exemple : \*** Date de naissance 31/14/99 ⇒ ‘’erreur’’. Car Mois : 1…12

 **\*** Age électeur = 16 ans ⇒ ‘’erreur’’. Car age électeur  18.

**Remarque :**

* + - Le contrôle de longueur peut être considéré comme un contrôle de.
		- Le contrôle de vraisemblance est souvent appelé contrôle de domaine ou contrôle de plage de valeur.
1. **Contrôle indirect :**

**Rappel :** Détermination de la justesse d’une information par rapport à d’autres informations.

* 1. **Contrôle de cohérence interne :**

C’est la vérification de la justesse de la valeur d’une partie d’une données par rapport aux autres parties de la même donnée. Il est utilisé surtout dans le cas des données à codification articulée.

**Exemple :**

* + 1. Num-etud :  ⇒ ( erreur car cycle = ‘’ C ‘’ incompatible avec AE= ‘’ 4 ‘’
		2. 31/02/1999 ⇒ erreur car Jour = 31 incompatible avec mois = ‘’ 02 ‘’
	1. **Contrôle de cohérence externe :**

Il consiste à vérifier la justesse de la valeur d’une donnée par rapport à d’autres données, soit dans le même fichier soit dans un autre fichier.

**Exemple :** Enregistrement étudiant

|  |  |  |  |
| --- | --- | --- | --- |
| **N°** | **Nom** | **Date de naissance** | **Date inscription** |
| 1 | A | 1999 | 1998 |

⇒ Erreur date de naissance =1999 < date inscription = 1998

**Remarque :**

1. On utilise souvent l’année en cour dans le contrôle de cohérence externe
2. Les contrôles de cohérence peuvent être représenté soit par des relations simples ( >, =, , <) soit par des calcules.

**Ordre d’exécution des contrôles :**

Contrôles directs

* Contrôle de présence
* Contrôle de type
* Contrôle de cadrage
* Contrôle de vraisemblance

Contrôles indirects

* Contrôle de cohérence interne
* Contrôle de cohérence externe

**Exercice :**

Dans un fichier étudiant représenté par : numéro, Nom, date de naissance, lieu de naissance, Numéro rue, Nom de la rue, Code postal, ville, téléphone, Année d’inscription, cycle d’étude, diplôme préparé, Année d’étude.

1. Proposer une codification pour le numéro d’étudiant qui tient compte de l’année d’inscription et qui permet de répartir les étudiants facilement en cycle court et cycle long
2. Représenter par un tableau l’ensemble des contrôles que peut subir chaque rubrique de ce fichier.