
UNIVERSITE SIDI MOHAMED BEN ABDELLAH
ECOLE SUPÉRIEURE DE TECHNOLOGIE DE FES

FILIEÈRE : TECHNIQUES DE GESTION
COMMERCIALE

Cours d'informatique

Information de base
Architecture d'un Ordinateur
Système d'exploitation (SE ou OS)
Introduction à la notation binaire
Présentation Internet
Langage HTML

Enseignant : Mr M.HMAMOU
Année Universitaire : 2015-2016

Chapitre1 : Information de base

1. Informatique

Le mot informatique a été proposé en 1962 pour désigner « le traitement automatique de l'information »

Informatique = **information** + automa**tique**

En 1966 l'académie française en a donné la définition suivante :

« Science du traitement rationnel, notamment par machines automatiques, de l'information considérée comme le support des connaissances humaines et des communications, dans les domaines techniques, économiques et sociaux ».

Science qui regroupe l'ensemble des théories et des techniques permettant de **traiter de l'information** à l'aide d'un ordinateur

Sciences et **T**echnologies de l '**I**nformation et de la **C**ommunication

Information : Aspect logique appelé **SOFTWARE** □ **Logiciels ou programmes**
Ordinateur : Aspect physique appelé **HARDWARE** □ **Matériels.**

2. L'ordinateur

Machine (**calculateur**) commandée par un **programme** enregistré qui permet de traiter des informations en exécutant une **séquence finie d'instructions** (opérations arithmétique et logique)

1.1 Fonctions principales

- recevoir des données : par les périphériques d'entrée comme le clavier.
- effectuer sur ces données des opérations : fonction de traitement réalisé à l'intérieur de l'unité centrale.
- Sauvegarder ou stocker ces données : dans les disques (disquette ou disque dur).
- restituer des résultats : grâce aux périphériques de sortie comme l'écran ou l'imprimante.

L'unité centrale, les périphériques d'entrées-sorties et les unités de disques constituent les organes de l'ordinateur. Ils sont reliés, fonctionnent de manière coordonnée, en exécutant un programme.

1.2 Caractéristiques d'un ordinateur

- Universel
- Rapide : Millions d'Instructions par seconde (MIPS)
- Fiable
- Grande Capacité mémoire

3. Algorithme

Un algorithme décrit la suite des actions à effectuer pour réaliser un traitement donné.

Un algorithme constitue donc une suite d'instructions finies (opérations) permettant à partir de données effectuer un traitement pour donner un résultat.

Un algorithme est énoncé dans un langage proche de l'utilisateur (langage naturel), il est Indépendant de tout langage de programmation

Ex : Si on souhaite calculer la moyenne de deux notes, l'algorithme est le suivant :

Début

```
Entrer note1;  
Entrer note2;  
Moyenne  $\leftarrow$  (note1 + note2) / 2;  
Sortir "moyenne = ", Moyenne;
```

Fin.

4. Programme

Un programme décrit une **procédure de calcul** qui à partir d'informations en entrée (**données**) produit des informations en sortie (**résultats**).

Une procédure de calcul est une **séquence** d'instructions d'un **langage de programmation**.

Un programme peut également être défini comme la traduction d'un algorithme dans un langage de programmation.

Un langage de programmation est un logiciel capable de communiquer les instructions de traitement à l'ordinateur.

Un processus est un programme en cours d'exécution.

Exemple : le programme Pascal qui permet de calculer la moyenne de notes :

```
Program Calcul_moyenne;  
Var  
    Note1 , note2 , Moyenne : Real;  
Begin  
    Write('donnez la note 1 : ');  
    Readln (note1);  
    Write('donnez la note 2 : ');  
    Readln (note2);  
    Moyenne := (note1 + note2) / 2;  
    Writeln('moyenne = ', Moyenne);  
End.
```

5. Logiciel

Ensemble de programmes, procédés et règles destinés à l'exploitation d'un ordinateur et son environnement.

Il existe deux types de logiciels :

5.1. logiciels de base :

□Système d'exploitation : comprend tous les programmes destinés à faciliter l'exploitation de la machine.

Ex : MS-DOS , Windows , Linux, ...

□Les utilitaires : il s'agit entre autres :

- De programme de tri ou fusion paramétrables qui peuvent être appelés par une application.
- De programme de manipulation de fichiers
- D'éditeurs de texte, véritables traitements de texte qui permettent l'écriture de programmes.

De plus, nous assistons actuellement à une généralisation de programmes offrant à l'écran une interface graphique sous forme d'aide et de menus déroulants.

5.2. logiciels d'application :

On distingue les logiciels de préparation de programme :

□Ceux qui permettent leur écriture : les éditeurs, les compilateurs, ...

□Ceux qui permettent leur conception : les générateurs d'écrans, de programmes ou de schémas.

Et les logiciels outils ou progiciels : ce sont des programmes conçues pour plusieurs utilisateurs potentiels.

6. Les programmes et leurs langages

Langage machine

- langage binaire
- Programme Exécutable

Langage d'assemblage

- Mnémoniques
- Programme Objet

Langage évolué

- Notation commune
- Programme source

exemples : C, JAVA, PASCAL...

Le langage machine et le langage d'assemblage dépendent du processeur (la puce)

Les langages évolués sont indépendants des machines

7. Fichier

Un fichier est un ensemble d'informations enregistrées sur la mémoire de l'ordinateur.

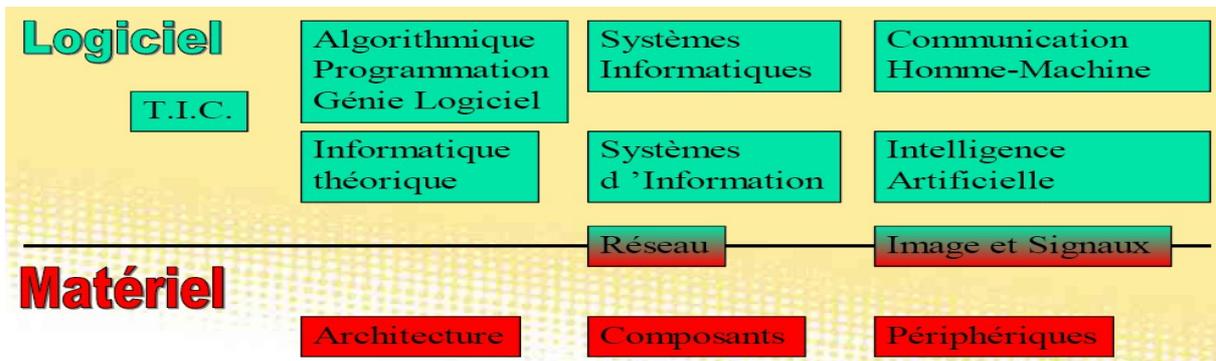
Un fichier est connu par son nom et une extension.

- Le nom d'un fichier est composé d'au maximum 8 caractères alphanumériques (sans l'espace et les caractères spéciaux).
- L'extension ou le suffixe (facultatif) est composé d'au plus trois caractères représentant le type du fichier.

On peut citer quelques extensions :

- * com : fichiers de commandes ex : command.com
- * exe : fichiers exécutables ex : find.exe
- * doc : fichiers document ex : lettre.doc

8. Les domaines de l'informatique



Chapitre 2 : Architecture d'un Ordinateur

1. Historique

Boulier chinois, 700

Machine de Pascal, 1642

Mémoire mécanique de Babbage, 1883

Machine à carte de Hollerith, 1890

Machines électroniques IBM, 1940

Von Neumann, 1946

Génération 1 (1945 - 1960)

- langage binaire
- Entrée : Carte perforée, Sortie :Imprimante
- 1000 opérations élémentaires/s

Génération 2 (1960-1965)

- Langage évolué
- transistors, diodes, mémoires à tores
- 100000 op. elem./s

Génération 3 (1965-1975)

- Circuit intégré, la puce
- 10⁹ à 10¹² op elem/s

Génération 4 (1975-)

- microprocesseur, dizaine de circuits sur une puce
- Développement des ordinateurs personnels

2. Familles d'ordinateur

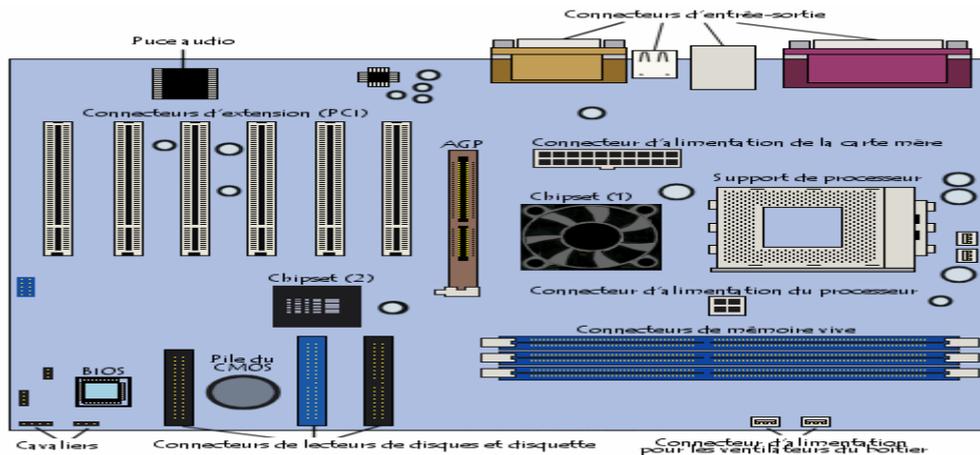
Ordinateurs centraux (MainFrame)

Ordinateurs Personnels (Personal Computer)

- Ordinateur de Bureau (Desktop computer)
- Ordinateur Portable (Laptop, Notebook)

Ordinateurs de poche/ assistants personnels (Personal Digital Assistant)

3. Organes d'un ordinateur



Comme son nom l'indique, la carte mère est une carte maîtresse, prenant la forme d'un grand circuit imprimé possédant notamment des connecteurs pour les cartes d'extension, les barrettes de mémoires, le processeur, etc.

5. L'unité centrale

L'unité centrale contient :

- l'alimentation : courant électrique.
- le microprocesseur : appelé cœur de l'ordinateur.
- l'horloge : cadence de travail du microprocesseur.
- Les bus de données : routes empruntées par les données qui transitent dans l'ordinateur entre ses différents composants.
- La mémoire centrale : mémoire principale

6. La mémoire centrale

C'est un dispositif de stockage des données.

Elle a pour fonction principale de stocker des informations afin d'être en mesure de les restituer le moment venu. Elle est placée sous les ordres de l'organe de commande (OC). Toute information ainsi stockée est accessible par l'intermédiaire d'une adresse contenue dans le registre d'adresse (ou registre de sélection mémoire) que lui attribue l'OC.

Pendant l'exécution d'un programme, la mémoire centrale contient généralement trois catégories d'informations : les éléments du système d'exploitation, les instructions du programme d'application en cours, et au fur et à mesure de leurs chargement, les données (ou opérandes) destinés à subir les traitements ainsi que celles qui en résultent.

Ensemble fini de cellules (ou case), Chaque case a un numéro : **adresse**

Unités de mesure: **bit**, octet (Byte), mot

Chaque case correspond à un mot.

La mémoire centrale est constituée de deux types :

☐ **La mémoire morte ROM** : Read Only Memory, mémoire morte, non volatile.

Elle est programmée par le constructeur et contient des programmes d'auto test et de vérification de l'état de fonctionnement de l'ordinateur. Sa tâche se termine juste après le démarrage de la machine.

- lecture seule
- instructions de démarrage
- quelques Ko

Exemple de la ROM : Le BIOS (Basic Input Output System) qui contient un programme permettant de configurer l'ordinateur pour le démarrage. Le BIOS est spécifique au type de la machine sur laquelle il tourne.

Une ROM bien connue est le **BIOS** accessible par son menu **setup** au démarrage de l'ordinateur.

Présentation du BIOS

Le **BIOS** (« *Basic Input/Output System* » traduisiez « Système de gestion élémentaire des entrées/sorties ») est un composant essentiel de l'ordinateur, permettant le contrôle des éléments matériels. Il s'agit d'un petit logiciel dont une partie est dans la ROM

Lorsque le système est mis sous-tension ou réamorcé (Reset), le BIOS fait l'inventaire du matériel présent dans l'ordinateur et effectue un test (appelé **POST**, pour "*Power-On Self Test*") afin de vérifier son bon fonctionnement.

- Effectuer un test du processeur (CPU)
- Vérifier le BIOS
- Vérifier la configuration du CMOS
- Initialiser le timer (l'horloge interne)
- Initialiser le contrôleur DMA
- Vérifier la mémoire vive et la mémoire cache
- Installer toutes les fonctions du BIOS
- Vérifier toutes les configurations (clavier, disquettes, disques durs ...)

Si jamais le POST rencontre une erreur, il va essayer de continuer le démarrage de l'ordinateur. Toutefois si l'erreur est grave, le BIOS va arrêter le système et :

- afficher un message à l'écran si possible (le matériel d'affichage n'étant pas forcément encore initialisée ou bien pouvant être défaillant) ;
- émettre un signal sonore, sous forme d'une séquence de bips (*beeps* en anglais) permettant de diagnostiquer l'origine de la panne ;
- envoyer un code (appelé code *POST*) sur le port série de l'ordinateur, pouvant être récupéré à l'aide d'un matériel spécifique de diagnostic.

Si tout est correct, le BIOS émettra généralement un bip bref, signalant qu'il n'y a pas d'erreur

□ **La mémoire vive RAM (Barettes) :** Random Access Memory, mémoire vive, Volatile.

C'est un dispositif physique capable d'enregistrer temporairement les informations, elle est caractérisée par sa capacité, son temps d'accès et sa volatilité.

- lecture et écriture
- volatile



- taille : centaines de Mo
- débit de transfert : 2 Go/s
- contient à la fois le programme à exécuter et ses données (**opération de chargement**)

7. La mémoire auxiliaire

Ce sont les périphériques de stockage et restitution d'information.

7.1 Unité de mesure :

Octet (Byte) : 8 bits

Ko (Kb)

Mo(Mb)

Go (Gb)

7.2 Les différentes mémoires auxiliaires :

7.2.1. disque dur : C'est une unité de stockage .

A la manière des tourne-disques d'autrefois, une tête de lecture se déplace au-dessus d'un disque de la périphérie vers le centre, pour lire et écrire des données.

A la différence près qu'il n'y a pas un disque mais *des* disques, chacun ayant sa propre tête de lecture (et d'écriture).

Lors de l'achat d'un nouveau disque dur, il peut être utile de s'intéresser :

_ à la capacité du disque dur, bien sûr mais attention ... certaines capacités ne sont pas supportées par les cartes mères !

_ au temps d'accès à l'information.

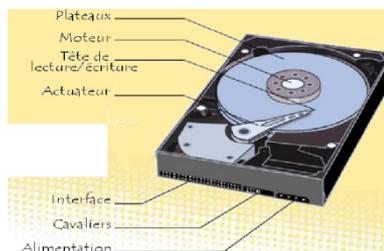
_ à la vitesse de rotation des disques ; plus elle augmente, plus le temps d'accès diminue.

Lecture et écriture

Taille : centaines de Go

Débit de transfert : 100 Mo/s

Logiciels + données



Formatage **Physique** (bas niveau)
Logique (haut niveau)

7.2.2. Disquettes

C'est une unité de stockage amovible.

Amovible car la disquette où sont stockées les informations peut se détacher du lecteur et ainsi se transporter facilement.

Son rôle est particulièrement utile lorsque le système d'exploitation est défaillant ; grâce à la disquette de démarrage, on peut tenter d'accéder au disque dur et essayer de relancer le système.

Cependant , il est de plus en plus supplanté par les clefs USB , supports amovibles qui se branchent sur le port du même nom et qui permettent d'enregistrer beaucoup plus de données.

lecture et écriture

taille : 1,44 Mo

débit de transfert : 150 ko/s

sauvegarde données (transport)

7.2.3. Clés USB

Une **clé USB** (en anglais *USB key*) est un périphérique de stockage amovible de petit format pouvant être branché sur le port USB d'un ordinateur.

Une clé USB embarque dans une coque plastifiée un connecteur USB et de la **mémoire flash**, une mémoire à semi-conducteurs, non volatile et réinscriptible, c'est-à-dire une mémoire possédant les caractéristiques d'une mémoire vive mais dont les données ne se volatilisent pas lors d'une mise hors tension.

Ainsi une clé USB est capable de stocker jusqu'à plusieurs gigaoctets de données, tout en étant capable de conserver les données lorsque l'alimentation électrique est coupée, c'est-à-dire lorsque la clé est débranchée.

Par ailleurs, les cartes mères récentes permettent de booter sur des clés USB, ce qui signifie qu'il est désormais possible de démarrer un système d'exploitation à partir d'une simple clé USB.

Les caractéristiques à prendre en compte lors du choix d'une clé USB sont les suivantes :

- **Capacité**
- **Taux de transfert** : il s'agit de la vitesse de transfert des données. Il est à noter que le taux de transfert en lecture est généralement différent du taux de transfert en écriture, car le processus d'écriture sur la mémoire flash est plus lent. Le taux de transfert dépend de la vitesse en lecture et en écriture de la mémoire Flash, ainsi que de la version de la norme USB supportée :
 - o USB 1.1 (USB bas débit) pouvant atteindre 12 Mbit/s,
 - o USB 2.0 (USB Haut débit ou Hi-speed) pouvant atteindre 480 Mbit/s. Il est important de noter que pour pouvoir profiter pleinement de ce taux de transfert, la clé devra être branchée sur un port USB 2.0. Dans le cas contraire (port USB 1.1), la clé fonctionnera à bas débit.
- **Fonctionnalités de chiffrement** : certaines clés proposent des outils permettant de **crypter** les données ou une partie des données présentes sur la clé afin d'en renforcer leur **confidentialité**.
- **Protection des données en écriture** : sur certaines clés, un interrupteur matériel permet de mettre la clé en lecture seule afin d'éviter la suppression ou la modification des données.

- **Fonctions multimédias** : lorsque la clé USB possède une prise casque et permet la lecture de fichiers audios (généralement au format **MP3**), on parle alors de lecteur MP3 ou baladeur MP3.

lecture et écriture

taille : centaines de Mo

débit de transfert : 1 Mo/s

sauvegarde données (transport)

7.2.4. CDROM, DVDROM

C'est un lecteur de disque optique.

Les têtes de lecture des unités précédentes lisent ou écrivent des données grâce à un flux électromagnétique.

La technologie de ce lecteur est différente ;

Un faisceau **laser** balaye la surface du disque, enduite d'une pellicule réfléchissante.

Cette surface n'est pas tout à fait lisse mais comporte des trous, à une échelle microscopique.

lecture

taille : 700 Mo (CD), 5 Go (DVD)

débit de transfert : 150 ko/s x Coefficient

logiciel, archives, multimédia

8. Le microprocesseur (MP ou μ P)

C'est le "*chef d'orchestre*" du PC .

Son rôle consiste à

- _ traiter et exécuter les instructions venant de programmes
- _ mettre en relation différentes parties de la carte.

Il constitue le cœur de l'ordinateur, endroit où s'effectuent les traitements. Il est composé de :

- Unité Arithmétique et logique (calcul)
- Unité de commande
 - Coordinateur
 - UAL
 - Mémoire
 - Périphériques
- Les registres : mémoires très rapides
- La mémoire cache

1.1 Fonctionnement de l'unité de commande

Organe de Commande : ou unité de contrôle, son rôle est d'extraire les instructions de la mémoire et de les analyser. Il doit assurer l'enchaînement logique des différentes instructions

d'un programme. A cette fin, il doit être en mesure de décoder ces instructions pour les transformer en ordres exécutables par sélection des circuits adéquats.

L'OC contrôle tous les circuits de l'UC de même que ceux assurant la liaison avec les matériels périphériques.

L'OC comporte deux registres :

- compteur d'instructions ou compteur ordinal, qui contient l'adresse de la nouvelle instruction à exécuter.
- Registre instruction, contient l'instruction extraite de la mémoire.

En résumé :

- Chercher en mémoire centrale l'instruction à exécuter et la stocker dans un registre
- décoder l'instruction
- Charger dans des registres les données nécessaires
- commander l'exécution par l'UAL
- récupérer le résultat dans un registre
- stocker le résultat en mémoire centrale
- recommencer pour l'instruction suivante

1.2 UAL

Unité Arithmétique et Logique ou unité de traitement, son rôle est d'effectuer tous les traitements arithmétiques et logiques.

Les opérandes qui lui sont fournis par l'OC, qui les recopie depuis la mémoire centrale dans des registres, constituent de véritables mini-mémoires spécifiques affectées à l'UAL.

1.3 L'horloge

C'est un bloc intégré au microprocesseur, elle détermine la cadence ou la vitesse de son traitement.

Cette horloge interne est donc un élément clé à prendre en considération lors de l'achat d'un ordinateur.

En effet, plus la *fréquence d'horloge* du μP est élevée plus le *nombre d'instructions exécutées par seconde par le processeur* est important. Cette fréquence s'exprime en Mhz, en Ghz ou en MIPS.

Ainsi, un PC cadencé à 500 Mhz traitera jusqu'à 500 millions d'instructions par seconde tandis qu'un PC cadencé à 2,5 Ghz pourra en traiter jusqu'à 2 milliards et demi par seconde !

L'intérêt d'une fréquence élevée est évident. Seul inconvénient : parallèlement, le μP chauffe ! C'est pourquoi il est équipé d'un radiateur thermique, qui dissipe la chaleur. De plus, la carte mère est généralement équipée d'un ventilateur qui améliore la circulation d'air dans le boîtier de l'unité centrale.

9. Les Bus

C'est le moyen de transport des données.

Les données (issues des périphériques ou du μ P) empruntent des *voies communes* physiquement matérialisées par des pistes en cuivre ou des nappes de fils.

Chacune de ces pistes (ou chacun de ces fils) travaille à une certaine *cadence* que l'on appelle la **fréquence de fonctionnement du bus** . Cette fréquence n'est en fait qu'une **fraction ou un multiple de la fréquence d'horloge du PC** et correspond au nombre de bits qui peuvent être transmis par seconde sur le bus.

Et si on considère le bus dans toute sa "largeur" , c'est-à-dire ;

_ non pas un mais tous les fils du Bus

_ non pas une mais toutes les pistes du Bus

On obtient alors la bande passante du Bus .

C'est le **débit du Bus** ; la **quantité totale de bits qu'il peut transporter, par seconde**. (On s'intéresse aux bits car les "données" transportées par les fils ou les pistes ne sont que des "0" ou des "1")

Le tableau suivant donne différentes formules de la bande passante d'un bus, en fonction de l'unité choisie :

| | | | | |
|--------------------------------|---|---|---|---|
| Bande passante (Bits/s) | = | Largeur (Bits) | x | Fréquence (Hz) |
| Bande passante (Octets/s) | = | $\frac{\text{Largeur (Bits)}}{8}$ | x | Fréquence (Hz) (car 1 octet équivaut à 8 bits) |
| Bande passante (Ko/s) | = | $\frac{\text{Largeur (Bits)}}{8 \times 1024}$ | x | Fréquence (Hz) (car 1 Ko équivaut à 1024 octets) |
| Bande passante (Mo/s) | = | $\frac{\text{Largeur (Bits)}}{8 \times 1024 \times 1024}$ | x | Fréquence (Hz) (car 1 Mo équivaut à 1024 Ko) |

En réalité chaque bus est généralement constitué de 50 à 100 lignes physiques distinctes, classées en trois sous-ensembles fonctionnels :

- Le **bus d'adresses** (appelé parfois *bus d'adressage* ou *bus mémoire*) transporte les adresses mémoire auxquelles le processeur souhaite accéder pour lire ou écrire une donnée. Il s'agit d'un bus unidirectionnel.
- Le **bus de données** véhicule les instructions en provenance ou à destination du processeur. Il s'agit d'un bus bidirectionnel.
- Le **bus de contrôle** (parfois *bus de commandes*) transporte les ordres et les signaux de synchronisation en provenance de l'unité de commande et à destination de l'ensemble des composants matériels. Il s'agit d'un bus directionnel dans la mesure où il transmet également les signaux de réponse des éléments matériels.

10. Les périphériques d'entrées / sorties

L'utilisateur communique avec l'ordinateur par l'intermédiaire des périphériques d'entrées. L'ordinateur communique à l'utilisateur au moyen de périphériques de sorties.

10.1. Les périphériques d'entrée

Assurent l'entrée d'informations dans l'UC. On peut citer :

10.1.1. Le clavier

C'est un périphérique d'entrée qui permet d'établir un dialogue "Homme-Machine".

Sa structure, très inspirée des machines à écrire, existe essentiellement en 2 formats :

- _ AZERTY pour les pays francophones
- _ QWERTY pour les pays anglophones

et permet, entre autres,

- _ de saisir des caractères alphanumériques ;
Touches [**A-Z**, **0-9**, etc...]

- _ d'intervenir sur le déroulement d'un programme ;
Touches [**Arrêt**, **Pause**, **Echap**, etc...]

- _ d'envoyer des commandes ;
Ainsi, taper simultanément les touches **Ctrl**, **Alt** et **Suppr** fait rebooter le PC.

Sur certains claviers évolués, une touche permet même de lancer un Navigateur Internet.

10.1.2. La souris

Il s'agit généralement d'un système mécanique monobloc.

Petit boîtier renferme une boule en contact à la surface sur laquelle il repose, qui glisse, grâce à une roue, sur un tapis.

Elle est reliée à l'unité centrale et dont le déplacement est reproduit à l'écran par le déplacement d'un pointeur (en général un flèche).

Son nom bizarre est dû au fait que ses larges boutons avant et sa forme profilée font penser, de façon un peu amusante, à une petite souris.

Son originalité ; tout déplacement de la souris sur le tapis est retranscrit à l'écran, par l'intermédiaire d'un curseur. Ce qui rend le PC plus convivial mais aussi interactif.

En effet, certaines applications "*réagissent*" au passage du curseur, en produisant des sons ou en changeant de couleur, par exemple.

Les menus des programmes deviennent plus accessibles et un clic droit sur la souris, fournit

une foule d'informations et de commandes, sur l'application en cours.
Le clic gauche, lui, permet de sélectionner voire même (si on le fait glisser) de déplacer des applications ou des icônes, considérées alors comme de simples objets virtuels !

10.1.3. Les scanners

Un **scanner** (mot français «numériseur») est un périphérique d'acquisition permettant de numériser des documents, c'est-à-dire de transformer un document papier en image numérique.

10.1.4. Les lecteurs optiques, ...

10.2. Les périphériques de sorties

Assurent la communication vers l'extérieur.

10.2.1 L'écran

C'est un périphérique de sortie, dédié à l'affichage.

Cet écran permet

- _ de visualiser et donc contrôler les données entrées au clavier
- _ de suivre le déroulement d'une ou plusieurs applications

Ses caractéristiques principales sont :

- _ son type : écran plat ou cathodique.

L'écran cathodique du PC est semblable à celui de notre bonne vieille TV ;

*Un faisceau d'électrons , guidé par un champ magnétique , vient heurter un point luminescent - **un pixel** - de la paroi interne de l'écran .*

Comme son intensité est modifiable et qu'il balaie rapidement l'écran, l'oeil a l'illusion de voir une image .

- _ Sa taille (en **pouces**) : C'est la longueur de la diagonale de l'écran .

(La diagonale étant *la ligne virtuelle qui va du coin supérieur gauche au coin inférieur droit*)

Plus grande est la taille de l'écran, meilleur est le confort visuel.

Malheureusement pour les écrans cathodiques, c'est synonyme d'encombrement.

- _ Sa résolution : C'est la quantité de pixels affichables à l'écran.

Elle s'exprime généralement sous la forme :

Nbre de pixels horizontaux x Nbre de pixels verticaux

De nombreuses résolutions sont utilisables.

Toutefois, il est conseillé d'adapter la résolution à la taille réelle de l'écran.

Pour un écran de 15 pouces, il est préférable d'utiliser une résolution de **800 x 600**.

_ sa fréquence de balayage : C'est la vitesse de balayage du faisceau d'électrons qui donne à l'oeil , l'illusion de voir une image ;

Plus la fréquence de balayage augmente , moins l'image clignote

_ son temps de réponse : Ce paramètre concerne les écrans plats .

Il diminue la "*traînée*" des images en mouvement et permet de rendre leurs mouvements plus fluides.

Il est donc important qu'il soit le plus faible possible, si on veut visionner des vidéos ou rendre ses jeux plus agréables à regarder.

10.2.2 L'imprimante

L'imprimante existe en plusieurs type (à jet d'encre, laser, ...)

L'imprimante est généralement caractérisée par les éléments suivants :

- **Vitesse d'impression** : exprimée en *pages par minute (ppm)*,
- **Résolution** : exprimée en *points par pouces* (notés *ppp* ou *dpi*, pour *dot per inch*), la résolution définit la finesse de l'impression. La résolution est parfois différente pour une impression monochrome, couleur ou photo.
- **Temps de préchauffage** : il représente le temps d'attente nécessaire avant la première impression.
- **Mémoire embarquée** : il s'agit de la quantité de mémoire permettant à l'imprimante de stocker les travaux d'impression.
- **Le format de papier** : selon leur taille, les imprimantes sont capables d'accueillir différentes taille de documents, généralement au format A4 (21 x 29,7 cm), plus rarement au format A3 (29,7 x 42 cm).
- **Alimentation papier** : il s'agit du mode de chargement de l'imprimante, caractérisant la façon de laquelle le papier vierge est stocké.
- **Cartouches** : les cartouches sont rarement standard et dépendent fortement de la marque et du modèle d'imprimante.

11. Les cartes d'extension

11.1. La Carte Son

Elle exerce 2 rôles :

_ Celui de **convertisseur analogique > numérique (CAN)** :

Au sens physique du terme , un son est un signal **analogique** .

C'est-à-dire qu'on peut le représenter à l'aide d'une courbe **continue**, variable dans le temps.

Ce signal est capté par la carte son grâce à ses entrées.

(Ce sont des connecteurs sur lesquels on peut brancher un micro, par exemple)

Cependant, un tel signal même de courte durée prendrait beaucoup de place s'il était stocké tel quel sur le disque dur.

C'est pourquoi le CAN :

▣ La découpe en "petits morceaux", à un intervalle de temps régulier qu'on appelle la **fréquence d'échantillonnage**.

▣ Ne garde que les échantillons les plus représentatifs et leur affecte un numéro.

(Plus ce numéro est grand, plus le signal stocké sera fidèle à l'original)

_ Celui de **convertisseur numérique > analogique (CNA)** :

Cette fois, il effectue l'opération inverse ; la restitution sonore.

À partir d'un fichier stocké sur le disque dur, il arrive à reconstruire le signal sonore qui est récupérable sur les sorties de la carte son.

(Ces sorties sont des connecteurs sur lesquels on peut brancher des enceintes, par exemple)

11.2. La carte Modem

Cette carte permet au PC de s'ouvrir véritablement au monde extérieur.

En effet, elle lui permet d'échanger des données avec un PC distant, via les lignes téléphoniques.

On peut ainsi :

- _ se connecter à Internet
- _ exporter ou importer des fichiers
- _ recevoir des messages

Certains modems sont même dotés d'une fonction **répondeur** ou **fax**.

Le principe utilisé pour transporter les données est celui de la modulation de signal .
MODEM signifie **MO**dulateur - **DE**Modulateur

C'est-à-dire qu'il transforme les données numériques (comme les fichiers stockés sur le disque dur) en un signal **analogique** (c'est-à-dire une courbe **continue** , variable dans le temps), véhiculable par les lignes téléphoniques.

Les caractéristiques principales d'une carte MODEM sont :

- _ ses différentes vitesses de transmission : (en bit/s ou bauds)
 - qui permettent de s'adapter à un MODEM distant moins performant ou à une ligne téléphonique de mauvaise qualité .

- _ La compression des données. Le MODEM est capable d'ignorer cette fonctionnalité si les données émises sont déjà compressées
- _ Le Rappel automatique ; c'est-à-dire la recombinaison du numéro.

11.3. La Carte Réseau

Cette interface permet d'intégrer notre PC à un **réseau**.

C'est-à-dire un *ensemble d'ordinateurs ou de périphériques externes - scanners , imprimantes ,etc ...- ayant des **communications à la fois physiques et logiques** .*

Les **communications physiques** représentent en fait les **supports de transmission**.

Elles peuvent être :

- _ en câble coaxial
- _ en paire torsadée
- _ en fibre optique
- _ ou sans fil

Les **communications logiques**, elles, représentent les **structures logiques du réseau**.

C'est-à-dire *les façons dont ses divers éléments communiquent entre eux*.

Elles peuvent être :

- _ en bus
- _ en anneau
- _ en étoile
- _ en arbre

Ainsi, il ne faut pas confondre ces 2 types de communications !

La carte réseau la plus connue est la carte **Ethernet** qui utilise la paire torsadée comme support de transmission et une structure en bus comme communication logique.

Mais il y en a d'autres, comme la carte **Token Ring** qui utilise une structure logique en anneau.

Pour notre PC, les avantages d'une carte réseau sont :

- _ sa rapidité
- _ sa facilité d'installation
- _ son faible coût
- _ la possibilité d'intégrer, de façon rapide, de nouveaux produits au réseau

Une fois dans le réseau, notre PC peut alors :

- _ partager des fichiers
- _ partager des imprimantes
- _ partager des applications
- _ effectuer des téléchargements
- _ utiliser des bases de données réparties

Chapitre 3 : Système d'exploitation (SE ou OS)

Logiciel indispensable : Tout autre logiciel communique via le SE avec l'ordinateur

Le SE est Chargé en mémoire centrale dès le départ.

Les SE les plus courants : WINDOWS, MAC OS, UNIX/LINUX

1. Rôle du SE

Gestion des différentes unités

Mémoire centrale

Processeur

Entrées/sorties

Mémoire auxiliaire

1. Gestion de la mémoire auxiliaire : Fichier et dossier

BIT (binary digit): Plus petite **unité d'information** manipulable par une machine numérique.

Deux valeurs possibles : 0 ou 1.

Fichier : **Unité de stockage** en mémoire auxiliaire. Il correspond à un ensemble d'information identifié par un nom unique. Sa taille se mesure en **octets** (8 bits).

2. Différents types de fichiers

Programmes, texte, graphique, image, son

Dossier / Répertoire / directory Ensemble de fichiers identifié par un nom.

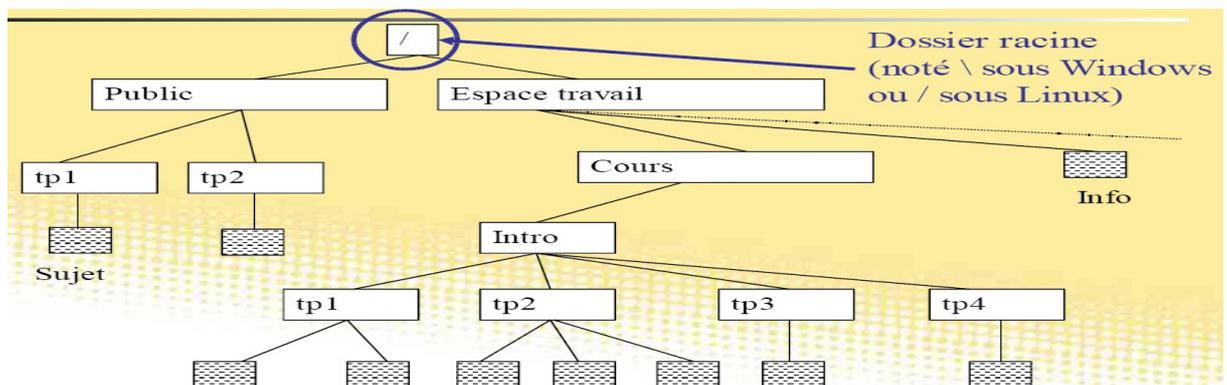
Espace de travail (home directory).

3. Gestion de la mémoire auxiliaire : Organisation

Le système d'exploitation permet de gérer l'organisation des dossiers et des fichiers sous forme d'une **arborescence**.

créer, supprimer, dupliquer, déplacer renommer...

4. Gestion de la mémoire auxiliaire : Arborescence

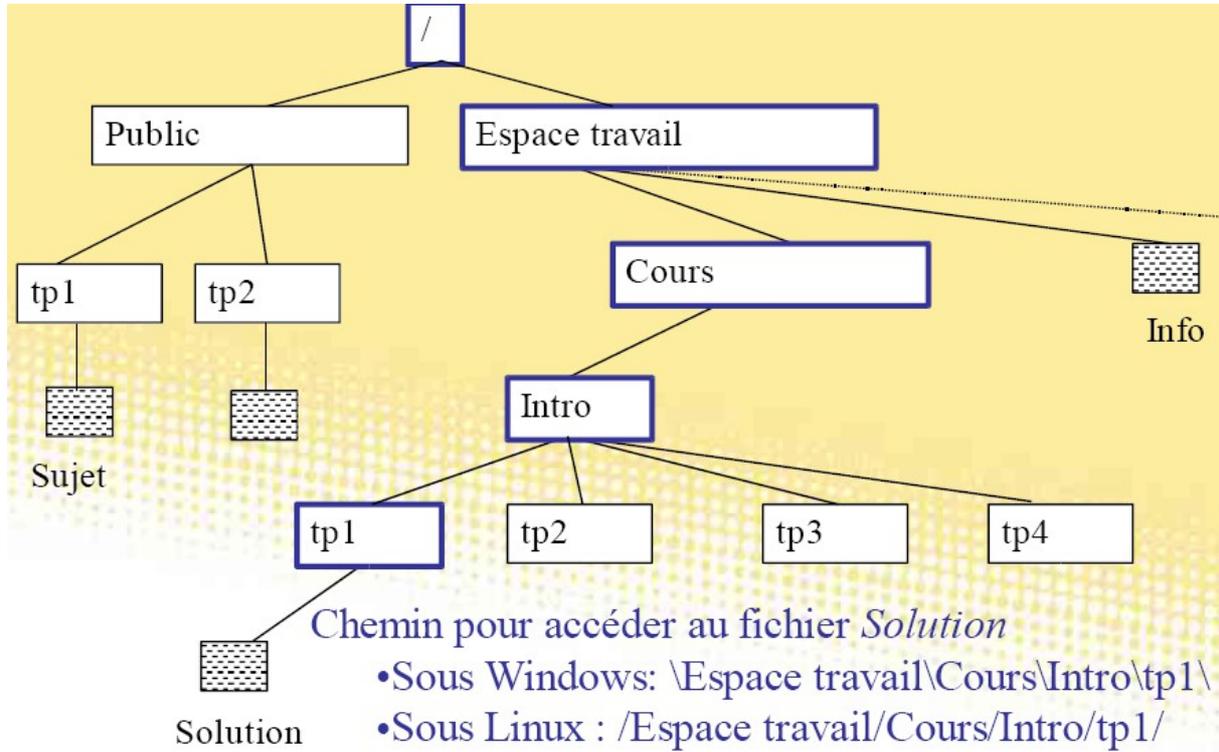


Les dossiers Nœuds : dossiers qui contiennent d'autres sous dossiers ou des fichiers.

Les fichiers sont les feuilles de l'arborescence.

5. Chemin

Succession de dossiers permettant l'accès à un fichier (ou un dossier) depuis la racine.



Chapitre 4 : Introduction à la notation binaire

1. Généralités

Vers la fin des années 30, Claude Shannon démontra qu'à l'aide de « contacteurs » (interrupteurs) fermés et ouverts pour qu'il était possible d'effectuer des opérations logiques.

contact fermé --> 1 --> «VRAI»

contact ouvert --> 0 --> «FAUX»

Deux états pour coder l'information numérique : Bit (binary digit)

1 bit = 2 états : 0 ou 1

2 bits = 4 états : 00, 01, 10, 11

3 bits = 8 états : 000, 001, 010, 011, 100, 101, 110, 111

n bits = 2^n états

La base binaire

2 chiffres : 0 et 1 : nombre binaire

Les unités standardisées (depuis 1998) :

- Un kilooctet (ko ou kB) = 1000 octets
- Un Mégaoctet (Mo ou MB) = 1000 ko = 1 000 000 octets
- Un Gigaoctet (Go ou GB) = 1000 Mo = 1 000 000 000 octets
- Un Téraoctet (To) = 1000 Go = 1 000 000 000 000 octets

2. Représentation de l'information de la base B à la base 10.

D'une manière générale un nombre N à la base B sera représenté (ou converti) à la base 10 de la manière suivante :

$$N = A_{n-1} * B^{n-1} + A_{n-2} * B^{n-2} + \dots + A_1 * B^1 + A_0 * B^0$$

Exemples :

$$2546 = 2*10^3 + 5*10^2 + 4*10^1 + 6*10^0$$

$$(01101101)_2 = 0*2^7 + 1*2^6 + 1*2^5 + 0*2^4 + 1*2^3 + 1*2^2 + 0*2^1 + 1*2^0$$

$$= 0 + 1*64 + 1*32 + 0 + 1*8 + 1*4 + 0 + 1*1$$

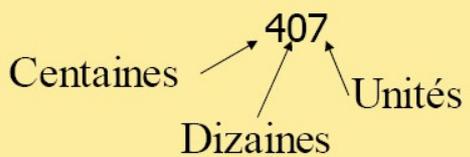
$$= 0 + 64 + 32 + 0 + 8 + 4 + 0 + 1 = 109$$

On représente d'une manière schématique un octet :

| | | | | | | | |
|-------|-------|-------|-------|-------|-------|-------|-------|
| 2^7 | 2^6 | 2^5 | 2^4 | 2^3 | 2^2 | 2^1 | 2^0 |
| | | | | | | | |

Changement de base (2 10)

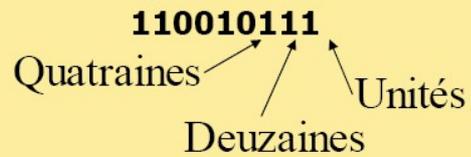
■ Base 10



$$= 4 * 100 + 0 * 10 + 7 * 1$$

- Unités : 10^0
- Dizaines : 10^1
- Centaines : 10^2
- Milliers : 10^3
- ...
- Millions : 10^6

■ Base 2



$$= 1 * 256 + 1 * 128 + 0 * 64 + 0 * 32 + 1 * 16 + 0 * 8 + 1 * 4 + 1 * 2 + 1 * 1$$

- $2^0 = 1$
- $2^1 = 2$
- $2^2 = 4$
- $2^3 = 8$
- $2^4 = 16$
- $2^5 = 32$
- $2^6 = 64$
- $2^7 = 128$
- $2^8 = 256$
- $2^9 = 512$
- $2^{10} = 1024$
- $2^{11} = 2048$

résultat : 407

3. Représentation de l'information de la base 10 à la base 2

Pour transformer un nombre à la base B on peut procéder par des divisions successives du nombre par B

Le nombre 109 est converti en base 2 de la manière suivante :

| | | | | | | | | | |
|-----|----|----|----|---|---|---|---|--|--|
| 109 | 2 | | | | | | | | |
| 1 | 54 | 2 | | | | | | | |
| | 0 | 27 | 2 | | | | | | |
| | | 1 | 13 | 2 | | | | | |
| | | | 1 | 6 | 2 | | | | |
| | | | | 0 | 3 | 2 | | | |
| | | | | | 1 | 1 | 2 | | |
| | | | | | | 1 | 0 | | |

$$109 \text{ (base 10)} = 01101101 \text{ (base 2)}$$

Changement de base (10 2)

■ Base 10

$$234 / 10 = 23 \text{ reste } 4$$

$$23 / 10 = 2 \text{ reste } 3$$

$$2 / 10 = 0 \text{ reste } 2$$

2 3 4

■ Base 2

$$234 / 2 = 117 \text{ reste } 0$$

$$117 / 2 = 58 \text{ reste } 1$$

$$58 / 2 = 29 \text{ reste } 0$$

$$29 / 2 = 14 \text{ reste } 1$$

$$14 / 2 = 7 \text{ reste } 0$$

$$7 / 2 = 3 \text{ reste } 1$$

$$3 / 2 = 1 \text{ reste } 1$$

$$1 / 2 = 0 \text{ reste } 1$$

résultat : 11101010

4. Arithmétique binaire

- Addition
 $0+0=0$, $0+1=1$, $1+0=1$, $1+1=0$ (et on retient 1)
 exemple :

$$\begin{array}{r} 11011 \quad (= 2^4+2^3+2^1+2^0 = 27) \\ + \quad 00110 \quad (= 2^2+2^1 = 6) \\ \hline = 100001 \quad (= 2^5+2^0 = 33) \end{array}$$

- Soustraction
 $0-0=0$, $0-1=1$ (et on retient 1) , $1-1=0$, $1-0=1$
 exemple :

$$\begin{array}{r} 111011 \quad (= 2^5+2^4+2^3+2^1+2^0 = 59) \\ - \quad 1110 \quad (= 2^3+2^2+2^1 = 14) \\ \hline = 101101 \quad (= 2^5+2^3+2^2+2^0 = 45) \end{array}$$

- Multiplication
 $0*0=0$, $0*1=0$, $1*0=0$, $1*1=1$
 exemple :

$$\begin{array}{r} 10011 \quad (= 2^4+2^1+2^0 = 19) \\ * \quad 101 \quad (= 2^2+2^0 = 5) \\ \hline 10011 \\ 00000 \\ 10011 \\ \hline = 1011111 \quad (= 2^6+2^4+2^3+2^2+2^1+2^0 = 95) \end{array}$$

- Division
 $0/0=$ indéterminé , $0/1=0$, $1/0=$ impossible , $1/1=1$

5. La base hexadécimale

Il existe deux autres bases très utilisées en informatique :

- la base octale (codage en base 8) : 0 .. 7
- la base hexadécimale (codage en base 16) : 0 .. 9 , A .. F

Comme il y a plus de chiffres hexadécimaux que de chiffres décimaux, le codage sera plus court. Le nombre 10 (décimal) s'écrit A en hexadécimal. L'avantage de l'hexadécimal est que chaque chiffre hexadécimal est composé de 4 chiffres binaires : "A" hexadécimal = "1010" binaire.

Pour convertir un nombre de la base 10 vers une base B, on procède par une division successive par B jusqu'à l'obtention de zéro.

Exemple : $(41235)_{10}$ à convertir en base 16.

$$\begin{array}{r} 41235 \quad 16 \\ 3 \ 2577 \quad 16 \\ \quad 1 \quad 161 \quad 16 \\ \quad \quad 1 \quad 10 \quad 16 \\ \quad \quad \quad 10 \quad 0 \\ \hline (41235)_{10} = (0A113)_{16} \end{array}$$

Voici les codes binaire et hexadécimal de quelques nombres

| Décimal | binaire | hexadécimal | Décimal | binaire | hexadécimal |
|---------|-----------|-------------|---------|-----------|-------------|
| 0 | 0000 0000 | 00 | 10 | 0000 1010 | 0A |
| 1 | 0000 0001 | 01 | 11 | 0000 1011 | 0B |
| 2 | 0000 0010 | 02 | 12 | 0000 1100 | 0C |
| 3 | 0000 0011 | 03 | 13 | 0000 1101 | 0D |
| 4 | 0000 0100 | 04 | 14 | 0000 1110 | 0E |
| 5 | 0000 0101 | 05 | 15 | 0000 1111 | 0F |
| 6 | 0000 0110 | 06 | 16 | 0001 0000 | 10 |
| 7 | 0000 0111 | 07 | 17 | 0001 0001 | 11 |
| 8 | 0000 1000 | 08 | 126 | 0111 1110 | 7E |
| 9 | 0000 1001 | 09 | 127 | 0111 1111 | 7F |

En réalité, sur un système 32 bits, comme la plupart des systèmes d'exploitation actuels (Windows, Mac OS, Linux...), le codage binaire et hexadécimal des nombres entiers se fait sur 32 bits, soit 4 octets (4 * 8 bits).

Et le nombre 1 s'écrira

0000 0000 0000 0000 0000 0000 0000 0001 en binaire

00 00 00 01 en hexadécimal

Les nombres négatifs

Pour simplifier, on supposera d'abord que notre système code les nombres sur 1 octet (comme dans le tableau précédent). Voici le code binaire et hexadécimal de quelques nombres négatifs :

| Décimal | binaire | hexadécimal | décimal | binaire | Hexadécimal |
|---------|-----------|-------------|---------|-----------|-------------|
| -1 | 1111 1111 | FF | -11 | 1111 0101 | F5 |
| -2 | 1111 1110 | FE | -12 | 1111 0100 | F4 |
| -3 | 1111 1101 | FD | -13 | 1111 0011 | F3 |
| -4 | 1111 1100 | FC | -14 | 1111 0010 | F2 |
| -5 | 1111 1011 | FB | -15 | 1111 0001 | F1 |
| -6 | 1111 1010 | FA | -16 | 1111 0000 | F0 |
| -7 | 1111 1001 | F9 | -17 | 1110 1111 | EF |
| -8 | 1111 1000 | F8 | -127 | 1000 0001 | 81 |
| -9 | 1111 0111 | F7 | -128 | 1000 0000 | 80 |
| -10 | 1111 0110 | F6 | | | |

Il y a une logique : tous les nombres négatifs commencent par 1 en binaire. On passe de 00 (0 en décimal) à 7F (127 en décimal), puis à 80 (-128 en décimal), jusqu'à FF (-1 en décimal).

| décimal | binaire | hexadécimal |
|---------|-----------|-------------|
| 0 | 0000 0000 | 00 |
| 1 | 0000 0001 | 01 |
| 127 | 0111 1111 | 7F |

| | | |
|------|-----------|----|
| -128 | 1000 0000 | 80 |
| -127 | 1000 0001 | 81 |
| -2 | 1111 1110 | FE |
| -1 | 1111 1111 | FF |

En réalité, sur un système 32 bits, le nombre -1 s'écrira

1111 1111 1111 1111 1111 1111 1111 1111 en binaire

FF FF FF FF en hexadécimal

Nombres signés et nombres non signés

Si les nombres sont signés (par défaut), sur 1 octet (8 bits) on peut coder les nombres de -128 à 127. Comme indiqué ci-dessus. Sur 4 octets (systèmes 32 bits), un nombre signé peut aller de -2 147 483 648 à 2 147 483 647.

Si les nombres sont non signés, sur 1 octet (8 bits), on peut coder les nombres de 0 à 255. Sur 4 octets (systèmes 32 bits), un nombre signé peut aller de 0 à 4 294 967 296.

Remarque : tout caractère utilisé est représenté par un nombre qui est codé en binaire. Ce nombre est le code ASCII (American Standard Code for Information Interchange).

Exemples : "A" = 65 "a" = 97
"B" = 66 "b" = 98

La touche ALT suivie du code ASCII permet de donner le caractère correspondant.

6. Travaux Dirigés

Exercice I :

Convertir en base décimale les entiers suivants :

- $(1011)_2$, $(1011001)_2$
- $(21)_8$, $(1067)_8$
- $(2AB1F)_{16}$, $(5DE)_{16}$

Convertir en base binaire : $(15)_{10}$, $(22)_{10}$, $(25)_8$ et $(A5D)_{16}$

Convertir l'entier 2397 successivement en base binaire, octal et hexadécimal.

Exercice II :

On utilise la technique de regroupement, convertir en base octal et hexadécimal : $(1100100100101)_2$, $(10010011110000101)_2$, $(1011010100010)_2$

Exercice III :

Combien d'entiers positifs peut-on coder en binaire sur un octet ? Combien de bits faut-il pour représenter 65563 entiers différents en binaire ?

Pour les deux exercices suivants, utilisez la table ASCII ci-dessous.

Exercice IV :

Donner la séquence binaire qui représente le message suivant : ALLO !

Décodez les séquences

debits : 1010011101010010011111010000, 1000111100110010101000110001

PS : ces séquences sont considérées comme des chaînes de caractères ASCII 7 bits.

Exercice V :

Ecrivez **Bonjour** en n'utilisant que la calculatrice, le clavier numérique et la touche Alt.

111

Exercice VI :

Effectuez ces additions en binaire

Vérifiez en décimal si vous ne vous êtes pas trompé(e)

1. $110011001 + 1101101$
2. $10111000 + 11000001$
3. $1111111 + 1111\ 11$
4. $1011 + 0101 + 1000 + 1100 + 1001 + 0100$
5. $1A34 + 34BC$
6. $1A34 + B4BC$
7. $5A34 + 64BC$
8. $AA34 + 94BC$

Exercice VII :

Effectuez ces soustractions en binaire

Vérifiez en décimal si vous ne vous êtes pas trompé(e)

| | | |
|-------------|------------|------------|
| 110011001 | 10111000 | 1111111 |
| - 1101101 | - 1001 | - 111111 |

Exercice VIII :

Effectuez ces multiplications en binaire

Vérifiez en décimal si vous ne vous êtes pas trompé(e)

| | | |
|-------------|----------|---------|
| 110011001 | 11000 | 1111 |
| * 101 | * 1001 | * 111 |

Chapitre 5 : Présentation Internet

1. Définition

En bref, Internet est le réseau mondial d'échange d'informations en tout genre...

Plus précisément, c'est un réseau international d'ordinateurs, ou plus précisément encore un réseau de réseaux d'ordinateurs, qui communiquent entre eux grâce à un protocole d'échange de données standard: TCP/IP (Transmission Control Protocol / Internet Protocol).

Chacun spécifie les droits d'accès à ses données

2. Histoire de l'Internet

- 1962 : Début de la recherche par ARPA, un projet du ministère de la Défense américain
- 1967 : première conférence sur ARPANET
- 1969 : connexion des premiers ordinateurs entre 4 universités américaines
- 1971 : 23 ordinateurs sont reliés sur ARPANET
- 1972 : naissance du InterNetworking Working Group, organisme chargé de la gestion d'Internet
- 1973 : l'Angleterre et la Norvège rejoignent le réseau Internet avec chacun 1 ordinateur
- 1979 : création des NewsGroups (forums de discussion) par des étudiants américains
- 1981 : apparition du Minitel en France
- 1982 : définition du protocole TCP/IP et du mot "Internet"
- 1983 : premier serveur de noms de sites
- 1984 : 1000 ordinateurs connectés
- 1987 : 10 000 ordinateurs connectés
- 1989 : 100 000 ordinateurs connectés
- 1990 : disparition d'ARPANET
- 1991 : Tim Berners-Lee invente le World Wide Web
- 1992 : 1 000 000 ordinateurs connectés
- 1993 : Netscape sort le navigateur "Mosaic"
- 1996 : 10 000 000 ordinateurs connectés
- 1999 : 200 000 000 utilisateurs dans le monde

Le Maroc compte plus de 4 millions d'internautes

3. Connexion à Internet

Quatre éléments doivent être pris en compte pour pouvoir se raccorder à Internet :

- Un ordinateur suffisamment puissant,
- Un modem ou un modem câble et/ou une carte réseau,
- Un compte ouvert auprès d'un fournisseur d'accès Internet (FAI),
- Des logiciels spécifiques de consultation des services Internet (navigateur,messagerie...).

Le modem (**modulateur/démodulateur**) est un appareil qui permet à votre PC de communiquer avec d'autres ordinateurs distants, via le réseau téléphonique.

Le modem câble est un appareil qui permet à votre PC équipé d'une carte réseau de communiquer, via le câble télévision.

Une carte réseau permet de relier l'ordinateur directement au réseau ou à un modem câble.

Un particulier accède à Internet via un prestataire de services, appelé fournisseur d'accès à internet (FAI ou "provider"). Ce dernier connecte l'ordinateur de chaque utilisateur sur le réseau.

Le Navigateur ou Browser « mot anglais » est un logiciel qui permet à l'utilisateur de rechercher, de consulter des documents et d'exploiter les liens hypertextuels qu'ils comportent.

(liste des synonymes de navigateurs ou Browser : furteurs, butineur, brouteur, arpenteur, fouineur)

Les principaux browsers sur le marché : Internet Explorer de Microsoft, Netscape Navigator et d'autres moins connus tel que : mozilla (multi-plateformes), Galeon, Konqueror, Safari(Apple),...

4. Le service Web

Il est impératif de différencier Internet, et le Web.

Ces deux mots sont très proches dans l'esprit de chacun, mais correspondent à deux notions très différentes.

Internet est un "réseau physique" (des ordinateurs reliés par des câbles).

Le "Web" (la toile d'araignée) est une application qui utilise le réseau Internet, et rend possible l'échange d'informations sur ce réseau, notamment grâce à l'utilisation des navigateurs, du langage HTML et donc du système de navigation hypertexte.

5. Les hébergeurs / serveurs web

Un hébergeur, ou serveur web, est simplement un ordinateur dédié à la diffusion de l'information.

Cet ordinateur est d'un côté connecté à Internet, et de l'autre, à une base de données spécifique.

Lorsqu'un utilisateur du réseau, ou client, tape une adresse web dans son navigateur, pour visualiser une page web, c'est en fait son navigateur qui envoie une requête au serveur, qui lui retourne la page correspondante (après certaines vérifications...).

6. Serveur web

Un service Web est un logiciel, dont le rôle est d'écouter des requêtes d'un type particulier (requête HTTP) provenant de *clients* que l'on appelle navigateurs. Les navigateurs sont bien connus des utilisateurs, et les plus répandus sont Netscape Navigator, Internet Explorer ou Opera.

Dans la suite, nous emploierons le terme de serveur pour désigner une machine qui propose de tels services, et nous appellerons donc serveur web une machine qui fait tourner un logiciel de service web.

Le protocole HTTP (Hyper Text Transfer Protocole) est assez simple : il permet au navigateur de demander à tout service web de lui retourner un fichier stocké sur le serveur. La plupart du temps, ces fichiers sont au format HTML (Hyper Text Format Language). Ce format consiste en du texte simple agrémenté de balises de mises en page qui permettent au navigateur de présenter le texte sous une forme plus agréable à lire.

Le service web doit être exécuté sur une machine qui possède une identification unique sous la forme d'un numéro IP (Internet Protocol), afin qu'un navigateur puisse localiser le service web sans ambiguïté. Ce numéro (ou adresse IP) est codé sur quatre octets, comme par exemple 192.67.78.90. Bien que seul le numéro IP soit nécessaire, la machine possède généralement un nom enregistré dans un DNS (Domain Name System), qui permet de retrouver l'adresse IP à partir du nom. Pour l'utilisateur humain, il est en effet plus simple de se souvenir d'un nom comme www.abc.fr que de 192.67.78.90.

Quand un service web tourne sur une machine (supposons 192.67.78.90), le protocole HTTP permet à un navigateur d'accéder à la page d'accueil du service web en indiquant comme adresse indifféremment

<http://192.67.78.90> ou <http://www.abc.fr>

On appelle cette adresse l'URL (Unique Resource Location)

7. Uniform Resource Locator

URL : le moyen de nommer un objet dans le monde WWW. La syntaxe d'un URL est la suivante :

protocole://serveur/répertoire/document.extension

Exemple <http://www.yahoo.fr>
[http:// agato.goglo.fr/ pub/](http://agato.goglo.fr/pub/)

Le type de serveur :

| | |
|---------------|--|
| http | : pour les URL provenant de serveurs WWW |
| gopher | : pour les serveurs GOPHER |
| ftp | : pour les fichiers à transférer |
| telnet | : pour ouvrir une session interactive |
| wais | : pour interroger une base WAIS |
| news | : pour accéder aux forums |

8. La recherche sur Internet

□ Les annuaires de recherche: pour les recherches thématiques: répertoires de sites classés par rubrique et indexés par des professionnels.

□ Les moteurs de recherche: pour les recherches à partir d'une liste de mots clés: robots qui collectent les adresses des sites et indexent automatiquement l'ensemble des pages Web.

9. Fonctionnement des moteurs de recherche

Ils recherchent l'information dans l'ensemble des pages référencées, et construisent une base de données de mots clés à partir de mots "significatifs" rencontrés:

- Mots clés associés aux pages
- Mots contenus dans le texte des pages
- Mots contenus dans les liens des pages
- Mots contenus dans les textes autour des liens vers les pages

10. Méthode pour mener une recherche sur Internet

1. dégager 1, 2 ou 3 mots clés;
2. les inscrire dans une thématique;

3. si le sujet de recherche peut faire l'objet d'un site Web, utiliser un annuaire;
4. sinon utiliser un moteur de recherche;
5. si la liste des réponses n'est pas satisfaisante, élargir ou resserrer la recherche, en s'inspirant des résultats obtenus lors de la recherche initiale;
6. puis utiliser les descriptifs proposés des sites pour sélectionner ceux qui semblent les plus pertinents.

11. Principes pour la liste des mots clés

1. séparer les mots clés par des espaces
2. pour faire une recherche sur un groupe de mots, utiliser "les guillemets"
3. une minuscule peut être transformée en majuscule, mais pas l'inverse
4. ajouter un + devant un mot assure la présence du mot
5. ajouter un - devant un mot exclut la présence du mot
6. ajouter un ~ entre 2 mots indique que les mots doivent être proches (moins de 10 mots entre les 2)

12. Autres services Internet

- La messagerie électronique
- Les dialogues en direct (Chat)
- Les forums de discussion
- FTP: le Protocole de Transfert de Fichiers
-

13. Travaux Dirigés

Exercice 1. Classes d'adresse

Quelles sont les classes des adresses réseaux suivantes ?

- 192.18.97.39 (www.javasoft.com) ;
- 138.96.64.15 (www.inria.fr) ;
- 18.181.0.31 (www.mit.edu) ;
- 226.192.60.40.

Pour chacune de ces classes, étant donné un réseau y appartenant, combien d'adresses de machines peuvent, a priori, être utilisées ?

Chapitre 5 : Langage HTML

1. Définitions

HTML est le langage de balisage hypertexte utilisé dans le Web (HyperText Markup Language).

Ce n'est pas un langage de programmation dans le sens où il n'existe pas de variables, boucles, expressions conditionnelles.

Ce sont "simplement" des balises pour mettre en forme (avec des liens, en tableau, etc...) du texte et des images.

Pour avoir une idée de l'aspect d'une page écrite en HTML, cliquez sur "Affichage" (ou "View") dans votre navigateur, puis sélectionnez "page source".

2. Besoins

Pour créer une page Web en HTML, nous avons besoin de:

1. Un navigateur (Internet Explorer, Netscape, ...), pour visualiser le résultat;
2. Un éditeur HTML (Notepad ou WordPad) pour créer votre page HTML;

à condition de sauver le fichier ainsi créé au format texte, avec une extension .html (ou .htm pour les versions de Windows antérieures à Windows 95).

NB: vous n'aurez pas besoin d'être connecté à Internet pour créer votre site.

3. Balise

Une balise est une "instruction" comprise entre crochets < > qui possède un nom et parfois des attributs.

 est, par exemple, la balise utilisée pour spécifier qu'il faut passer à la ligne.

La plupart des balises doivent être ouvertes puis refermées (début et fin).

Ouverture : <nom_de_balise>
Fermeture : </nom_de_balise>.

Par exemple: mon texte en gras (bold).

Le nom de la balise (contenu entre les crochets) n'est pas sensible à la casse : il peut être écrit indifféremment en majuscule, en minuscule ou en un mélange des 2.

L'attribut d'une balise est défini comme suit: nom_attribut="valeur"

Par exemple, la balise utilisée pour faire un lien a pour nom "a", et pour nom d'attribut "href". Ainsi, Google permet de faire un lien vers la page principale du site de Google.

Dans ce cas, la valeur de l'attribut "href" est donc "http://www.google.fr".

4. Structure d'une page HTML

<html> première balise d'un fichier HTML

```
<head>  
<!-- entête --- !>  
<title>Mon titre</title>  
<!-- titre --- !>  
</head>
```

```
<body>
```

```
<!-- corp : tout le document à afficher contenant texte, images,... --> !>
Voici mon premier document!
</body>
</html> dernière balise d'un fichier HTML
```

Le document doit être sauvegardé avec comme extension ".htm" ou ".html", et comme type "texte seulement" ou "tout type (*.*)";

Un double-clic sur le document : le navigateur par défaut sera alors ouvert sur la page.

Remarque :

```
<!-- ceci est un commentaire dans mon fichier HTML -->
<!-- il n'apparaîtra donc pas dans le texte de mon document -->
```

5. Balises simples

: passage à la ligne;
: mettre un texte en gras (bold);
<i>: mettre un texte en italique;
<u>: souligner un texte (underline);
<center>: centrer un texte;
etc...

Exemple :

```
<center> <b> <u> Mon texte </u> </b> <br> Et la suite... </center>
```

on obtient :

Mon texte
Et la suite...

6. Balises de mise en forme

6.1. La balise de couleur

```
<font color="red">Mon texte</font>
<font color="#FF0000">Mon texte</font>
```

Le premier exemple est utilisable pour quelques couleurs prédéfinies: red, green, black...
Sinon, on peut utiliser le code commençant par #, basé sur le système "RGB" (Red-Green-Blue): les 2 premiers chiffres (en base 16, donc allant de 0 à F) fournissent le taux de rouge dans la couleur, les 2 suivants le taux de vert, et les 2 derniers le taux de bleu.

Par exemple, si vous voulez obtenir du rose, mélange du rouge et du bleu, tapez Mon texte

Choisissez vos couleurs à partir du "colour picker" suivant ...

6.2. La balise de taille

Taille absolue (de 1 à 7):

```
<font size=1>Mon texte</font>
<font size=2>Mon texte</font>
<font size=3>Mon texte</font>
<font size=4>Mon texte</font>
<font size=5>Mon texte</font>
```

`Mon texte`

`Mon texte`

Ou taille relative:

`Mon texte`

`Mon texte`

`Mon texte`

`Mon texte`

6.3. Les listes

`` correspond aux listes numérotées (Ordered List);

`` aux listes à puces (Unordered List);

et `` sert à spécifier qu'on démarre un nouvel élément de liste;

à noter que `` existe également, mais n'est pas nécessaire, car supposé à l'ouverture du

`` suivant, ou à la fermeture de liste ``

Exemple :

``

` Ligne 1`

` Ligne 2:`

``

` Sous-Ligne 2.1`

` Sous-Ligne 2.2`

``

``

On obtient:

□ Ligne 1

□ Ligne 2

1. Sous-Ligne 2.1

2. Sous-Ligne 2.2

7. Les tableaux

7.1. Balises simples de tableau

<TABLE> et </TABLE> servent respectivement à ouvrir une nouvelle table, et fermer la table courante;

<TR> et </TR> servent à définir une ligne du tableau;

<TD> et </TD> servent à définir une cellule du tableau (à noter que </TD> n'est pas indispensable, pour la même raison que).

<TH> et </TH> servent à définir un titre de ligne ou de colonne (le texte apparaît alors centré en gras dans la cellule correspondante);

et <CAPTION> et </CAPTION> servent à définir un titre pour le tableau.

Exemple :

```
<table>
<caption>Mon titre</caption>
<tr>
<th>Nom 1
<th>Nom 2
</tr><tr>
<td>Cellule 1
<td>Cellule 2
</tr>
</table>
```

On obtient:

Mon titre
Nom 1 **Nom 2**
Cellule 1 Cellule 2

7.2. Options de la table:

border: spécifie l'épaisseur du cadre extérieur, en pixels. Avec la valeur 0, le cadre est invisible;

cellspacing: épaisseur en pixels autour de chaque cellule;

cellpadding: épaisseur en pixels entre l'élément de la cellule et le cadre;

width: largeur occupée par le tableau en pixels ou en pourcentages;

align: l'alignement du tableau: center, right ou left.

Exemple

En remplaçant la balise <table> de l'exemple précédent par: <table border=1 cellpadding=2 cellspacing=5>, on obtient:

Mon titre

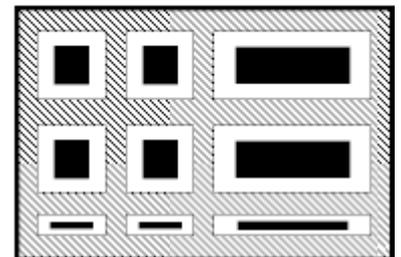
| Nom 1 | Nom 2 |
|-----------|-----------|
| Cellule 1 | Cellule 2 |

7.3. Options des cellules:

<td colspan=2>: fusion horizontale de 2 cellules;

<td rowspan=2>: fusion verticale de 2 cellules.

Table border 



Cellspacing 

Cellpadding 

Cell content 

8. Les formulaires

- 1 balise <FORM> ... </FORM>
 - 2 attributs
 - METHOD
 - GET ou POST
 - indique le mode de transfert des données
 - choisir POST
 - ACTION
 - définit la localisation du script permettant d'exploiter le formulaire

```
<FORM METHOD="POST" ACTION="/cgi-bin/inscript">
```

Nom:

Prénom:

Profession

Homme

Femme

Message:

- **classe 2** : SELECT => utilisé pour création de listes

- **menus déroulants**

```
<SELECT NAME = "semaine" >
```

```
<OPTION> lundi
```

```
<OPTION> mardi
```

```
<OPTION SELECTED> mercredi
```

```
</SELECT>
```

Mercredi ▼
Lundi
Mardi
Mercredi
Jeudi
Vendredi
Samedi
Dimanche

- **listes à ascenseur**

```
<SELECT NAME = "menu" SIZE = "6" MULTIPLE>
```

```
<OPTION> cict
```

....

</SELECT>



– classe 3 : TEXTAREA

<TEXTAREA NAME = "commentaires" ROWS="10" COLS="5">

Entrez vos commentaires </TEXTAREA>

Nom:

Prénom:

Profession

Homme

Femme

9. Images

Attributs:

align: alignement désiré de l'image (de même que pour les tableaux)

alt: texte associé à l'image (apparaît quand la souris passe au dessus, ou en remplacement de l'image si celle-ci n'existe pas)

width: longueur

height: hauteur

10. Les liens hypertextes

Texte du lien

Liens à l'intérieur d'une même page:

Placer: à l'intérieur de votre document

Texte du lien désigne alors l'emplacement désigné

à utiliser par exemple en bas de page pour proposer de revenir en haut de page.

Les chemins d'accès peuvent être:

absolus: c'est-à-dire qu'on spécifie l'adresse complète de ce à quoi on fait référence:

par exemple C:/images/monImage.gif pour une image, ou http://www.google.fr pour un site;

relatifs: la référence est faite par rapport à l'endroit où l'on se trouve:

par exemple ./monImage.gif ou ./maPage.html désignant des fichiers situés dans le même répertoire que le fichier qui y fait référence.

Dans ce cas:

"." désigne le répertoire courant,
et ".." désigne le répertoire parent.

Par exemple, dans votre répertoire nommé "site" se trouve un répertoire nommé "images" et un fichier nommé "index.html".

Dans le fichier "index.html", nous voulons faire référence à l'image nommée "monImage.gif" du répertoire "images".

La référence se fait alors comme suit: ``

Si nous sommes dans un fichier du répertoire "images" et nous voulons faire référence au fichier "index.html", la référence se fait comme suit: `Ma ref`

11. Options de la balise body

Les principales options de la balise body:

`bgcolor=couleur`: couleur de fond

`background=chemin_image`: image de fond

`text=couleur`: couleur du texte

`link=couleur`: couleur des liens

`vlink=couleur`: couleur des liens visités

`alink=couleur`: couleur des liens actifs (au moment où on clique dessus)

12. Organisation du texte

12.1. Les paragraphes

`<p> paragraphe</p>`

`<p align=justify>paragraphe</p>`

`<p align=left>paragraphe</p>`

`<p align=center>paragraphe</p>`

`<p align=right>paragraphe</p>`

12.2. Les titres

de `<H1>` à `<H6>`:

`<H1>Mon texte</H1>`

`<H2>Mon texte</H2>`

`<H3>Mon texte</H3>`

`<H4>Mon texte</H4>`

`<H5>Mon texte</H5>`

`<H6>Mon texte</H6>`

11.3. Les séparateurs

`<HR>`

Exemple : `<HR width="50%">`

12.4. Les caractères spéciaux

Les caractères tels `<` et `>` étant interprétés par votre navigateur comme marqueurs de balises, Pour cela, il existe les caractères spéciaux. Ceux-ci commencent par le signe "&" et finissent par un point-virgule ";".

Ainsi, par exemple, il faut taper `>` pour obtenir le signe `>`.

Cette liste est également utile pour afficher des caractères n'existant pas sur notre clavier.

Ainsi, par exemple, pour obtenir le caractère €, il faut écrire `€`

13. Les frames

`<FRAMESET>` et `<FRAME>`

Exemple

`<HTML>`

`<HEAD>`

```

<TITLE>Frames</TITLE>
</HEAD>
<FRAMESET COLS="250,*">
<FRAME SRC="/menu.html">
<FRAME SRC="/page1.html" NAME="pagePrincipale">
<NOFRAMES>
<H1>No Frames? No Problem!</H1>
Take a look at our <A HREF="/page1.html">no-frames</A> version.
</NOFRAMES>
</FRAMESET>
</HTML>

```

Référence entre pages

Si maintenant nous voulons qu'en cliquant sur un lien de la page "menu", ce soit la page "pagePrincipale" qui soit modifiée, nous écrivons:

```
<a href="maReference" target="pagePrincipale">
```

14. Map, area et images réactives

```


<map name="mymap">
<area shape="rect" coords="0,0,100,50" href="page1.html" alt="zone 1">
<area shape="rect" coords="100,0,200,50" href="page2.html" alt="zone 2">
<area shape="rect" coords="200,0,300,50" href="page3.html" alt="zone 3">
</map>

```

3 types de formes peuvent être utilisées :

la forme rectangulaire, comme dans l'exemple : shape="rect" → coords="left-x, bottom-y, right-x, top-y"

la forme sphérique : shape="circle" → coords="center-x, center-y, radius"

ou la forme polygonale : shape="poly" → coords="x1, y1, x2, y2, ..., xN, yN"

Les principales balises HTML

Mise en forme des caractères

| | |
|--------------------------------------|--|
| ... | Texte en gras |
| <BIG>...</BIG> | Agrandissement de la taille des caractères |
| <BLINK>...</BLINK> | Texte clignotant (Netscape seul) |
| ... | Texte en italique |
| ... | Texte en couleur où XXXXXX est une valeur hexadécimale |
| ... | Taille des caractères où X est une valeur de 1 à 7 |
| <I>...</I> | Texte en italique |
| <NOBR>...</NOBR> | Empêche les ruptures automatiques de ligne des navigateurs |
| <PRE>...</PRE> | Texte préformaté, soit avec affichage de tous les espaces et sauts de ligne |
| <SMALL>...</SMALL> | Réduction de la taille des caractères |
| ... | Mise en gras du texte |
| _{...} | Texte en indice |
| ^{...} | Texte en exposant |
| <U>...</U> | Texte souligné |

Mise en forme du texte

| | |
|------------|--------------------------------------|
| <!--...--> | Commentaire ignoré par le navigateur |
|------------|--------------------------------------|

| | |
|--|--|
| | A la ligne |
| <BLOCKQUOTE>... </BLOCKQUOTE> | Citation (introduit un retrait du texte) |
| <CENTER>...</CENTER> | Centre tout élément compris dans le tag |
| <DIV align=center> ...</DIV> | Centre l'élément encadré par le tag |
| <DIV align=left> ...</DIV> | Aligne l'élément à gauche |
| <DIV align=right> ...</DIV> | Aligne l'élément à droite |
| <Hx>...</Hx> | Titre où x a une valeur de 1 à 6 |
| <Hx align=center>...</Hx> | Titre centré |
| <Hx align=left>...</Hx> | Titre aligné à gauche |
| <Hx align=right>...</Hx> | Titre aligné à droite |
| <P>...</P> | Nouveau paragraphe |
| <P align=center>...</P> | Paragraphe centré |
| <P align=left>...</P> | Paragraphe aligné à gauche |
| <P align=right>...</P> | Paragraphe aligné à droite |

Listes

| | |
|--------------|-------------------------------------|
| | Liste non numérotée (dite à puces) |
| | Élément de liste |
| | |
| | Liste numérotée |
| | Élément de liste |
| | |
| <DL> | Liste de glossaire |
| <DT>...</DT> | Terme de glossaire (sans retrait) |
| <DD>...</DD> | Explication du terme (avec retrait) |
| </DL> | |

Ligne de séparation

| | |
|-------------------|--------------------------------------|
| <HR> | Trait horizontal (centré par défaut) |
| <HR width="x%"> | Largeur du trait en % |
| <HR width=x> | Largeur du trait en pixels |
| <HR size=x> | Hauteur du trait en pixels |
| <HR align=center> | Trait centré (défaut) |
| <HR align=left> | Trait aligné à gauche |
| <HR align=right> | Trait aligné à droite |
| <HR noshade> | Trait sans effet d'ombrage |

Hyperliens

| | |
|-------------------------------|--|
| ... | Lien vers une page Web |
| ... | Lien vers une adresse eMail |
| ... | Lien vers la page locale fichier.htm située dans le même répertoire |
| ... | Définition d'une ancre |
| ... | Lien vers une ancre |
| ... | |

Images

| | |
|-----------------------------|--|
| | Insertion d'une image au format Gif ou Jpg |
| | (voir liens pour l'adressage) |
| | Mise à l'échelle de l'image en pixels (a comme effet d'accélérer l'affichage de la page) |
| | Définition de la bordure d'une image avec lien |
| | Texte alternatif lorsque l'image n'est pas affichée |
| | Aligne l'image en bas |
| | Aligne l'image au milieu |
| | Aligne l'image en haut |
| | Aligne l'image à gauche |
| | Aligne l'image à droite |
| | Espacement horizontal entre l'image et le texte |
| | Espacement vertical entre l'image et le texte |

Tableau

| | |
|---|--|
| <code><TABLE>...</TABLE></code> | Définition d'un tableau |
| <code><TABLE width="x%"></code> | Largeur du tableau en % |
| <code><TABLE width=x></code> | Largeur du tableau en pixels |
| <code><TABLE border=x></code> | Largeur de la bordure |
| <code><TABLE cellpadding=x></code> | Espace entre la bordure et le texte |
| <code><TABLE cellspacing=x></code> | Epaisseur du trait entre les cellules |
| <code><TR>...</TR></code> | Ligne du tableau |
| <code><TD>...</TD></code> | Cellule du tableau |
| <code><TD bgcolor="#XXXXXX"></code> | Couleur d'une cellule de tableau |
| <code><TD width="x%"></code> | Largeur de colonne en % |
| <code><TD width=x></code> | Largeur de colonne en pixels |
| <code><TD align=center></code> | Texte dans la cellule centré |
| <code><TD align=left></code> | Texte dans la cellule aligné à gauche |
| <code><TD align=right></code> | Texte dans la cellule aligné à droite |
| <code><TD valign=bottom></code> | Alignement vers le bas du contenu d'une cellule |
| <code><TD valign=middle></code> | Centrage vertical du contenu d'une cellule |
| <code><TD valign=top></code> | Alignement vers le haut du contenu d'une cellule |
| <code><TD colspan=x></code> | Nombre de cellules à fusionner horizontalement |
| <code><TD rowspan=x></code> | Nombre de cellules à fusionner verticalement |

Frames

| | |
|---|---|
| <code><FRAMESET>...</FRAMESET></code> | Définit une structure de frames (remplace alors le tag BODY) |
| <code><FRAMESET rows="x%,y%,..."></code> | Division horizontale de la fenêtre en % |
| <code><FRAMESET cols="x%,y%,..."></code> | Division verticale de la fenêtre en % |
| <code><FRAME src="fichier.htm"></code> | Fichier affiché dans une fenêtre de frames |
| <code><NOFRAMES>...</NOFRAMES></code> | Contenu pour les browsers non prévus pour les frames |

Fichier Html

| | |
|--|--|
| <code><HTML>...</HTML></code> | Début et fin de fichier Html |
| <code><HEAD>...</HEAD></code> | Zone d'en-tête d'un fichier Html |
| <code><TITLE>...</TITLE></code> | Titre affiché par le browser (élément de HEAD) |
| <code><BODY>...</BODY></code> | Début et fin du corps du fichier Html |
| <code><BODY bgcolor="#XXXXXX"></code> | Couleur d'arrière-plan (en hexadécimal) |
| <code><BODY background="xyz.gif"></code> | Image d'arrière-plan |

15. Les feuille de style CSS

[ici](#) le lien vers les supports de cours de Fabien Torre sur les feuilles de style CSS.

Et [ici](#) la feuille de style utilisée pour cette page.

Principe

L'idée de l'utilisation de feuilles de style (en anglais *Cascading Style Sheets (CSS)*) est de dissocier le contenu d'un document de sa mise en forme.

Les intérêts de cette séparation sont nombreux :

un style, appliqué à toutes les pages d'un site, assure la cohérence graphique de ce site ;

la maintenance d'un site en est facilitée, un changement de charte graphique ne portera que sur la feuille de style et la modification sera naturellement propagée à toutes les pages ;

le code HTML est lisible et facilement modifiable, sans que l'on ait à penser en même temps au contenu et à son apparence ;

les pages, dégagées des balises de mise en forme, sont plus légères, donc plus rapides à circuler sur un réseau (la feuille de style est, le plus souvent, très légère et ne sera chargée qu'une seule fois pour l'ensemble des documents qui l'utilisent) ;

les balises donnent une indication sur la nature du contenu et cette indication pourra être utilisée par un moteur de recherche.

La feuille de style est donc un fichier *externe* à vos pages HTML. Ce fichier doit avoir comme extension *.css*

Et pour spécifier qu'une page HTML utilise une certaine feuille de style (qu'on nomme ici *myStyle.css*), on écrit ceci dans l'en-tête de la page HTML (balises *head*):

```
<link rel="stylesheet" type="text/css" href="myStyle.css">
```

Remarque: les navigateurs gèrent souvent une même feuille de style de manière différente...

Après avoir écrit une feuille de style, il conviendra donc de vérifier l'apparence du document avec un maximum de navigateurs.

Enfin, et surtout, il faut s'assurer impérativement que le document reste lisible même sans la feuille de style.

Utilisation

La syntaxe générale dans une feuille de style est la suivante:

```
nomClasse{  
nomAttribut1: valeur1;  
nomAttribut2: valeur2;  
etc...  
}
```

Les commentaires sont inclus entre */** et **/*

Par exemple:

```
H1{  
font-size: 20pt;  
text-align: center;  
text-decoration: underline;  
color: red;  
}
```

va permettre de définir que tout titre H1:

est de taille 20 (*pt = point*),

est centré

est souligné,

et est de couleur rouge.

Liste des propriétés CSS

Une liste complète des propriétés CSS est accessible [ICI](#).

Voici une liste des plus utilisées: on présente ici d'abord le nom de l'attribut, sa signification, puis la liste de ses valeurs possibles:

Propriétés de police:

font-size: taille de police:

numérique (10pt ...) / xx-small / x-small / small / medium / **large** / x-large / **xx-large**

font-family: famille de police:

Arial / Helvetica / Times New Roman / Sans Serif ...

font-style: style de police:

normal / *italic* / *oblique*

font-weight: graisse de police:

normal / **bold** / et de 100 (très fin) à **900 (très gras)**

font-variant: variante de police:

normal / SMALL-CAPS (PETITES CAPITALES)

color: couleur de police:

red / #FF0000 ...

Mise en forme du texte:

text-decoration: décoration du texte:

underline / overline / ~~line-through~~ / blink / none

text-transform: transformation du texte:

capitalize (début des mots en majuscule) / UPPERCASE / lowercase / none

Agencement du texte:

display: position par rapport au reste: nouveau paragraphe (block) / ou à la suite du paragraphe précédent (inline)

text-align: alignement horizontal: à gauche (left) / centré (center) / à droite (right) / justifié (justify)

text-indent: retrait du texte: numérique

line-height: interligne: numérique

vertical-align: alignement vertical: aligner en haut (top) / aligner au milieu (middle) / aligner en bas (bottom) / aligner sur la ligne de base, ou en bas s'il n'y a pas de ligne de base (baseline) / mettre en indice, sans réduire la taille de la police (sub) / mettre en exposant, sans réduire la taille de la police (super) / aligner sur le bord supérieur de l'écriture (text-top) / aligner sur le bord inférieur de l'écriture (text-bottom)

Mise en forme de listes:

list-style-type: type de représentation:

numériques: decimal / lower-alpha / upper-alpha / lower-roman / upper-roman

à puces: disc / circle / square

list-style-position: retrait des listes:

puces et numérotation dans le corps de la liste (inside) / retrait à gauche des puces et numérotation (outside)

list-style-image: graphique de puces personnalisées:

url(chemin_vers_image.gif)

Couleurs et arrière-plans:

color: couleur du texte

background-color

background-image

background-repeat: effet de répétition (repeat, repeat-x, repeat-y, no-repeat)

background-attachment: effet de filigrane (scroll, fixed)

background-position

Retraits extérieurs et intérieurs:

margin (retrait extérieur): numérique

margin-top

margin-bottom

margin-left

margin-right

padding (retrait intérieur)

padding-top

padding-bottom

padding-left

padding-right

Cadres:

border: numérique

border-top

border-bottom
border-left
border-right
border-style (solid, double, groove, ridge, inset, outset, dotted, dashed)
border-style-top
border-style-bottom
border-style-left
border-style-right
border-width
border-color
Positionnement absolu des éléments:
Balises div et span:
position (static, absolute, relative)
top
left
width
height
visibility (visible, hidden)
z-index (Position de la couche en cas de superposition)
Pseudo-classes pour liens hypertextes:
A:link
A:hover (passage de souris)
A:active (click de souris)
A:visited (prioritaire sur les précédents... ne pas utiliser mais préférer le VLINK de la balise BODY)

Exemples

Pour définir de façon globale la mise en forme du corps du document:

```
body{  
background-color: #CCFFCC;  
font-size: 11pt;  
color: #000000;  
font-family: Helvetica, Arial, sans-serif;  
}
```

Pour définir de façon globale la mise en forme des liens hypertextes:

```
A:link{  
text-decoration: none;  
color: #0000FF;  
background-color: inherit;  
}
```

Exercices sur les feuilles de style

Modifier vos pages précédemment créées en utilisant les feuilles de style.

Pour cela, vous pouvez utiliser les interfaces écrites par Dominique Gonzalez et récupérer les différents styles pour les inclure dans vos propres pages :

appliquer des styles aux [textes](#) ;

aux [listes](#) ;

aux [tableaux](#).

Projet

Un site personnel devant contenir au minimum: une page d'accueil, un CV, et une page personnelle.

Quelques critères d'évaluation:

les pages créées doivent être claires et agréables;

le style du site doit être cohérent, et le parcours facile;

le code qui l'a généré doit également être le plus clair possible;

plus vous utiliserez d'éléments du cours, plus je serai sûr que vous avez bien assimilé ces éléments...

La syntaxe CSS de base

Supposons que nous voulions que le fond d'une page Web soit d'un beau rouge :

Avec HTML, nous l'aurions fait comme ceci :

```
<body bgcolor="#FF0000">
```

Avec CSS, on peut obtenir le même résultat comme cela :

```
body {background-color: #FF0000;}
```

Comme vous l'aurez remarqué, les codes sont plus ou moins identiques pour HTML et CSS. Cet exemple illustre également le modèle fondamental de CSS :

```
selector {property: value;}
```

↑
À quelle(s)
balise(s) HTML
la propriété
s'applique
(p.ex. "body")

↑
La propriété, par
exemple, la couleur
d'arrière-plan
("background-color")

↖ La valeur de la
propriété de couleur
d'arrière-plan, par
exemple, le rouge
("#FF0000")

Mais où place-t-on le code CSS ? C'est précisément ce que nous allons voir maintenant.

Appliquer CSS à un document HTML

Il y a trois façons d'appliquer le style CSS à un document HTML. Elles sont toutes expliquées ci-dessous. Nous vous recommandons d'examiner la troisième méthode, c'est-à-dire celle externe.

Méthode 1 : Dans la ligne (l'attribut style)

Une façon d'appliquer du style CSS à HTML est celle avec l'attribut HTML `style`. En reprenant l'exemple précédent avec l'arrière-plan rouge, on peut l'appliquer ainsi :

```
<html>  
<head>  
  <title>Exemple</title>  
</head>  
<body style="background-color: #FF0000;">  
  <p>Cette page est rouge</p>  
</body>  
</html>
```

Méthode 2 : Interne (l'élément style)

Une autre méthode consiste à inclure le code CSS avec la balise HTML `<style>`. Par exemple, comme ceci :

```
<html>  
<head>  
  <title>Exemple</title>  
  <style type="text/css">  
    body {background-color: #FF0000;}  
  </style>  
</head>  
<body>  
  <p>Cette page est rouge</p>  
</body>  
</html>
```

Méthode 3 : Externe (un lien vers une feuille de style)

La méthode recommandée est celle avec un lien vers ladite feuille de style externe. Pour le reste de ce tutoriel, nous utiliserons cette méthode pour tous les exemples.

Une feuille de style externe est simplement un fichier texte ayant l'extension « .css ». Comme n'importe quel fichier, la feuille de style peut être rangée sur un serveur Web ou sur un disque dur.

Par exemple, disons que votre feuille de style se nomme « style.css » et se trouve dans un dossier appelé « style ». On peut illustrer cette situation comme ceci :



L'astuce consiste à créer un lien depuis le document HTML (default.htm) vers la feuille de style (style.css). Ce lien peut être créé en une ligne de code HTML :

```
<link rel="stylesheet" type="text/css" href="style/style.css" />
```

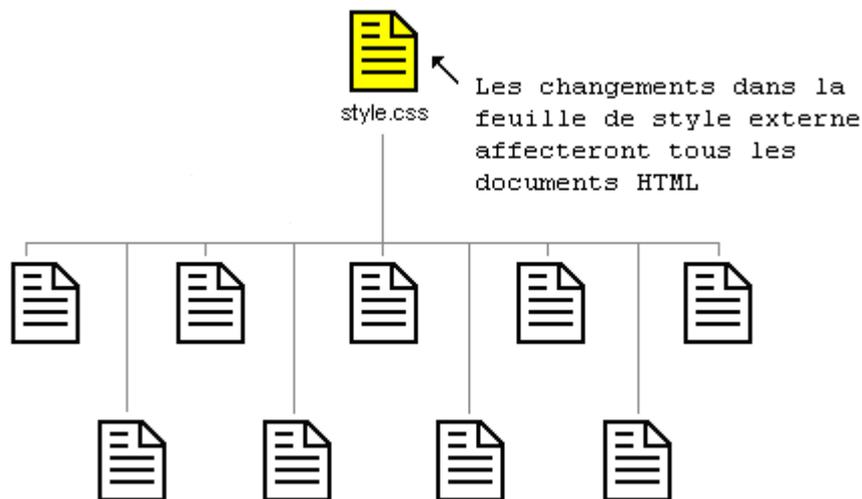
Remarquez comment on indique le chemin de la feuille de style avec l'attribut href.

La ligne de code doit s'inscrire dans la section d'en-tête du code HTML, c'est-à-dire entre les balises <head> et </head>. Comme ceci :

```
<html>
<head>
    <title>Mon document</title>
    <link rel="stylesheet" type="text/css" href="style/style.css" />
</head>
<body>
...

```

Ce lien instruit le navigateur d'utiliser la présentation du fichier CSS pour afficher le fichier HTML. Ce qui est vraiment fûté, c'est que plusieurs documents HTML peuvent être reliés à la même feuille de style. En d'autres termes, on peut utiliser un seul fichier CSS pour contrôler la présentation de plusieurs documents HTML.



Des documents HTML liés à la même feuille de style

Cette technique est susceptible d'économiser beaucoup d'efforts. Par exemple, si vous voulez changer la couleur d'arrière-plan d'un site Web contenant 100 pages, une feuille de style peut vous épargner la modification manuelle des 100 documents HTML. Avec CSS, la modification peut intervenir en quelques secondes en changeant juste une seule ligne de code de la feuille de style centrale. Mettons donc en pratique ce que nous venons d'apprendre.

Essayez vous-même

Lancez Bloc-notes (ou votre éditeur de texte habituel) et créez deux fichiers, un fichier HTML et un fichier CSS, avec les contenus suivants :

default.htm

```
<html>
<head>
    <title>Mon document</title>

```

```
        <link rel="stylesheet" type="text/css" href="style.css" />
    </head>
    <body>
        <h1>Ma première feuille de style</h1>
    </body>
</html>
```

style.css

```
body {
    background-color: #FF0000;
}
```

Placez maintenant les deux fichiers dans le même dossier. Songez à sauvegarder les fichiers avec les bonnes extensions (respectivement « .htm » et « .css »)

Ouvrez « default.htm » dans votre navigateur et constatez le fond rouge de la page.

Félicitations ! Vous avez fabriqué votre première feuille de style !

15. Travaux Dirigés :

Partie 1 : Structure d'une page, titres, paragraphes

Listes, texte préformaté

EXERCICE 1:

Réaliser la page Web simple illustrée par la copie-écran ci-dessous. Utilisez pour ce faire les 2 outils suivants :

- ouvrez le Bloc-notes Windows dans la moitié de gauche de l'écran
- lancez le browser Internet Explorer dans la moitié de droite de l'écran

kaoutar EL OUAZZANI

Mes coordonnées

kaoutar EL OUAZZANI
Av soussa VN
FES

Mon curriculum vitae

J'ai fréquenté l'école primaire de ...

J'ai suivi ensuite ...

Mes loisirs

à remplir...

kaoutar EL OUAZZANI / révision 13.04.09

Enregistrez sous exo1.html

Exercice 2:

I. Réalisez la liste à puce suivante :

Liste à puce

- Adresse : Av Soussa BD Elmouahidinne
- Tél : (212)667859646
- Fax : (212)667859647
- E-mail : h.merzouk@hotmail.com

Enregistrez sous exo21.html

II. Réalisez la combinaison de listes numérotés et à puce suivante :

Liste numérotée et à puce

1. Ecole primaire :
 - 1ère : Oum el khoura
 - 2e-3e : Lazrak
2. Collège :
 - 1ère-3e : Benabdellaziz

- 4e-6e scient. : Imam Malekh
- 2. Lycée :
 - IBN HAITAM
- 2. Université :
 - Sidi Mohamed Ben Abdellah, FST-FES

Enregistrez sous exo22.html

III. Réaliser la liste descriptive suivante

Liste descriptive

2004

Certificat de fin d'études secondaires (Baccalauréat)

2006

Diplôme DUT Option: GAI

2007

Licence Professionnelle Management International

Enregistrez sous exo23.html

IV. Affichez le petit dessin ci-dessous (Texte préformaté)

petit dessin

```

      ||
    /==||====\
   /             \
  /=====\<
 |  _  _  |
 | |  |  | |

```

Enregistrez sous exo24.html

Exercice 3 (Facultatif) : Donner le code html du plan suivant

- Filières EST

1. Electricité

- a. Electronique
- b. Electrotechnique
- c. Reseaux et telecom

2. Informatique

- développement
- Administration Réseaux
- Sécurité

3. Management

- Filières offshoring

- 1. Administration du Personnel
- 2. Techniques de Ventas
- 3. Administration réseaux

Partie2 : Liens hypertextes, Tableaux, Frames(cadres)

Exercice 1 :

Reprenez la page HTML créée lors du premier exercice et réalisez-y les compléments décrits ci-dessous :

- 1. Ajoutez au début de la page (juste sous le titre de premier niveau "Kaoutar EL OUAZZANI") une table des matières hypertexte de ce genre pointant vers des "cibles" que l'on mettra en place au point suivant de l'exercice :

1. [Mes coordonnées](#)
2. [Mon curriculum vitae](#)
3. [Mes hobbies](#)

2. Mettez en place les "cibles" (des liens créés au point précédent) juste avant les titres de chapitre, puis testez que vos renvois hypertextes de table des matières fonctionnent

Exercice 2 :

Réaliser le tableau ci-dessous :

| A.U 2008/2009 | Etudiants Techniques de Management | |
|-------------------------|------------------------------------|-------------|
| | 1ère année | 2ème année |
| Nombre total d'étudiant | à compléter | à compléter |
| Nombre de modules | à compléter | à compléter |

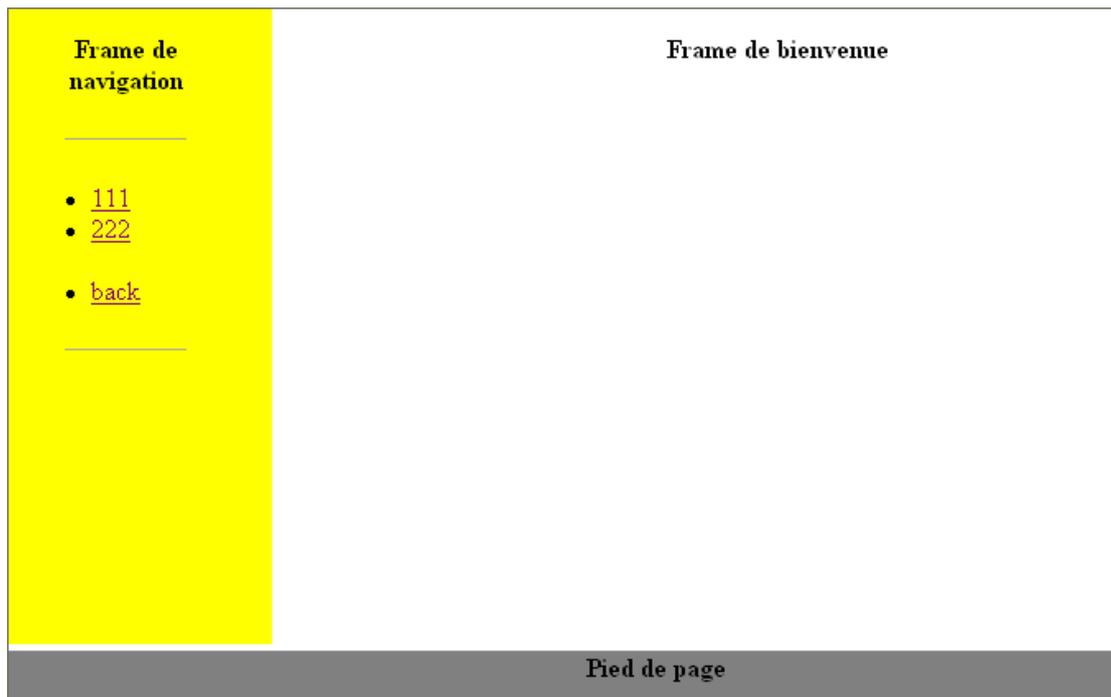
Tableau 1 : Effectifs

Utilisez des couleurs de fonds différentes pour les cellules du tableau.

Exercice 3 :

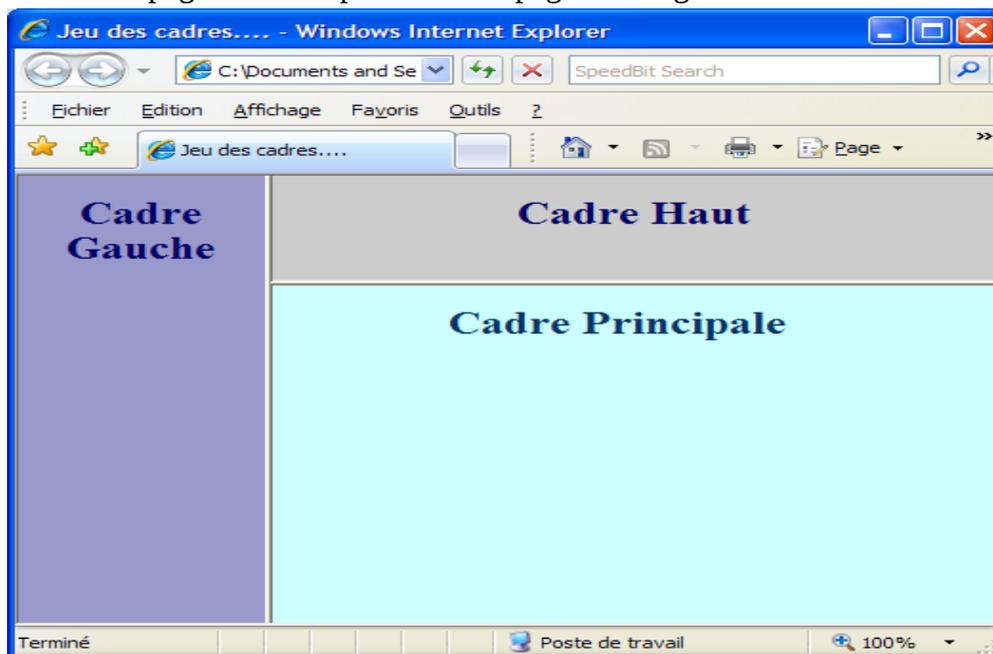
Réaliser la fenêtre suivante contenant des frames :

- la zone de "pied de page" doit avoir 30 pixels de haut, et cette hauteur doit pas être modifiable ; cette zone ne doit pas avoir d'ascenseurs
- la zone de "navigation" doit avoir 160 pixels de large, et cette largeur doit pouvoir être modifiable
- les zones de "navigation" et de page proprement dite doivent avoir des ascenseurs si nécessaire
- les bordures entre frames doivent avoir 2 pixels de large
- donnez des couleurs de fond différentes à chacun de ces frames
- lorsque l'on clique sur "111" ou "222" dans le frame de navigation, la page correspondante doit s'afficher dans la zone de page ; par contre lorsque l'on clique sur "back" la page correspondante doit remplir toute la fenêtre Netscape
- prévoir le cas où le browser attaquant ce frameset n'implémente pas le mécanisme des frames en renvoyant sur la page de "navigation"



Exercice 4 (Facultatif):

Ecrire une page HTML reproduisant la page de la figure suivante.



Correction :

Fichier Frameset:

```
<HTML><HEAD><TITLE>Déf frameset</TITLE></HEAD>
<FRAMESET cols="100,*" BORDER="4">
  <FRAME SRC="frame_navig.html" NAME="navig">
  <FRAMESET Rows="60,*">
    <FRAME SRC="frame_haut.html" NAME="haut" NORESIZE SCROLLING="no">
    <FRAME SRC="frame_page.html" NAME="page">
  </FRAMESET>
</FRAMESET>
</HTML>
```

Fichier Frame navig :

```
<html><body><center><B>Cadre Gauche</B></center> </body></html>
```

IDEM pour Fichier Frame_page Fichier Frame_haut

Exercice 5(Facultatif) :

Créez le code source de cette frame

Correction :

```
<FRAMESET ROWS=" *, *, * ">
  <FRAME SRC="doc F.html">
  <FRAMESET COLS="3*, * ">
    <FRAME SRC="docA.html">
    <FRAME SRC="docB.html">
  </FRAMESET>
  <FRAMESET COLS=" *, *, 2* ">
    <FRAME SRC="docC.html">
    <FRAME SRC="docD.html">
    <FRAME SRC="docE.html">
  </FRAMESET>
</FRAMESET>
```