



Matière:

Bureautique & Technologies

WEB

(Semestre 1)

Responsable :

Mr BENDAHMANE M.F.



Semestre 1

UNITE FONDAMENTALE (UEF 1)

- Mathématiques 1 [6 crédits]
- Physique 1 [6 crédits]
- Chimie 1 [6 crédits]

UNITE METHODOLOGIE (UEM 1)

- TP Physique 1 [2 crédits]
- TP Chimie 1 [2 crédits]
- **Bureautique et Web** [**3 crédits**]

UNITE DECOUVERTE (UED 1)

- Physique et ses applications [2 crédits]
- Environnement [2 crédits]

UNITE CULTURE GENERALE (UCG 1)

- Français 1 [1 crédit]

VHG: 345H équivalent à 30 crédits



Déroulement des enseignements

✚ 1H30 de cours / semaine

✚ 1H30 de TP / semaine

VHG: 45H équivalent à 3 crédits

Moyenne de la matière = (note du C.C. + 2 x E. F.) / 3

Moyenne \geq 10.00: VOUS OBTENEZ 3 CREDITS

Moyenne $<$ 10.00: Système de compensation



Systeme d'évaluation

1. Contrôle Continu :

- 6 points pour la présence. **JUSTIFIER VOS ABSENCES!!**
- 14 points sur le travail accompli lors des séances des TP.
(Noter par les enseignants des TP).

Total: 20 points

2. Examen Final :

- QCM (questionnaire à choix multiples)
- 1 ou 2 exercices.



Programme du Semestre 1



Partie 1: Bureautique

1. Le monde merveilleux de l'informatique.
2. Systèmes d'exploitation.
3. MS-Office 2003: WORD.
4. MS-Office 2003: Excel.
5. MS-Office 2003: Power Point.



Partie 2: Technologies WEB

6. Technologies Web: Services Web
7. Le langage HTML.



Objectifs de la formation

A la fin de cette formation vous devrez être capable de :

- Rédiger un rapport, une lettre ou un CV en utilisant le logiciel WORD.
- Faire des graphiques et des courbes en utilisant EXCEL.
- Faire des calculs suivant des formules et des relations mathématiques.
- Faire une présentation en utilisant Power Point.
- Créer une page web.



Matière: Informatique (Semestre 2)

Responsable :

Mr BENDAHMANE M.F.



Semestre 2

UNITE FONDAMENTALE (UEF 2)

- Mathématiques 2 [6 crédits]
- Physique 2 [6 crédits]
- Chimie 2 [6 crédits]

UNITE METHODOLOGIE (UEM 2)

- TP Physique 2 [2 crédits]
- TP Chimie 2 [2 crédits]
- **Informatique** [**5 crédits**]

UNITE CULTURE GENERALE (UCG 2)

- Histoire des Sciences [2 crédit]
- Français [1 crédit]



Programme du Semestre 2

1. Concepts de base.
2. Règles générales d'écriture d'un programme en langage Pascal.
3. Les structures de contrôles.
4. Les tableaux.
5. Le traitement des chaînes de caractères.
6. Les types définis par l'utilisateur – Type ensembles.
7. Les fonctions et les procédures.
8. Les fichiers de types textes.



Objectifs de la formation

A la fin de cette formation vous devrez être capable de :

- Maîtriser l'algorithmique, base de toute programmation.
- Maîtriser le langage Pascal.
- Écrire un programme à partir d'une relation ou formule mathématique.



BON COURAGE !



Cours 1 :

Le Monde Merveilleux de l'Informatique

Mr. Arbaoui Bachir



C'est quoi un Ordinateur (computer)??

+ Quelques définitions:

- C'est un appareil électronique et numérique qui stocke et traite de l'information.
- C'est une machine programmable qui accepte les données, les traite et fait sortir les résultats.
- C'est un dispositif qui fait appel à la technologie numérique et manipule les informations.



Que fait un ordinateur?

- Une machine traitant l'information fonctionne en convertissant tous types d'information en nombres binaires (uns et zéros), ensuite en utilisant de simples mathématiques de prendre des décisions, ou à réorganiser, ces nombres

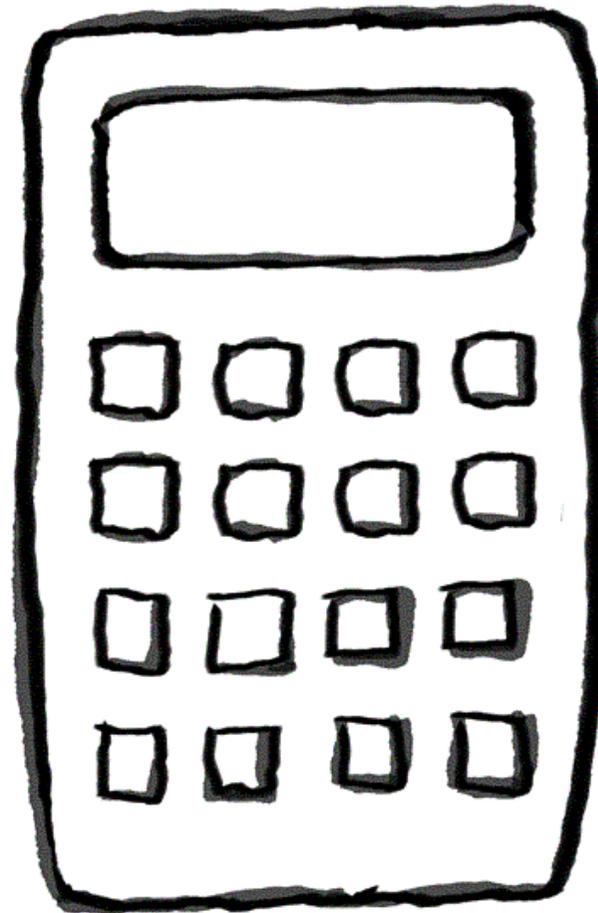


✚ Il y a deux choses essentielles pour comprendre le fonctionnement de l'ordinateur:

- La première est qu'un ordinateur traite tout type d'informations (pas uniquement des chiffres, mais aussi des lettres, des mots, des dates, et ainsi de suite) comme si c'était simplement des binaire de uns et de zéros. Par exemple, un ordinateur peut traduire la lettre "A" tapé sur le clavier dans une chaîne de uns et de zéros, comme 1000001.
- La deuxième est que les fonctions de l'ordinateur sont basées sur le mouvement et la transformation des impulsions électriques (représentant les uns et les zéros) dans des circuits électriques. À l'intérieur de l'ordinateur se sont ces circuits électriques qui effectuent les calculs sur les zéros et de uns, tels que les additionner ou les soustraire.



En gros, un ordinateur est une grande.....





Un peu d'histoire

- ✚ Les ordinateurs ont été mis au point par l'homme 'A-J.', comme moyen de calculer de grandes séries de nombres.
- ✚ Le temps n'a développé que la sophistication des ordinateurs, c-à-d:
 - Taille
 - Vitesse
 - Logiciels et applications
 - Prix



La préhistoire

- ✚ L'abaque a été inventé en l'an 300 A.J par les babyloniens; il est encore utilisé aujourd'hui dans certains pays asiatiques.
- ✚ L'abaque est un dispositif mécanique utilisé pour faire des calculs mathématiques





Le moyen âge

- ✚ 1617 - L'écossais John NEPER (1550-1617) inventât les logarithmes. En effet, celui-ci démontra que la multiplication et la division pouvaient se ramener à une série d'additions. Ceci permit l'utilisation de la règle à calcul.
 - Pour autant le vrai père de la théorie des logarithmes est « Mohamed Ybn Moussa Al-KHAWAREZMI », un savant arabe issu de la ville persane appelée « Khawarezm ». Ce savant développa par ailleurs l'Algèbre, terme provenant de l'arabe « Al-Jabr », qui signifie compensation, sous-entendu « la compensation par la recherche de la variable inconnue X afin d'équilibrer les résultats des calculs ».

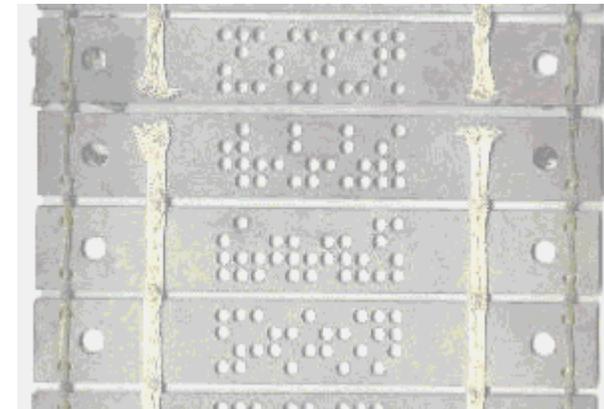


- ✚ 1623 - William Schickard inventa la première machine à calculer mécanique.
- ✚ 1643 - Blaise Pascal créa la machine d'arithmétique (baptisée Pascaline), une machine capable d'effectuer des additions et soustractions, destinée à aider son père, un percepteur de taxes.





- ✚ 1673 - Gottfried Wilhelm Von Leibniz ajouta à la Pascaline la multiplication et la division.
- ✚ 1801 - le Français Joseph-Marie Jacquard invente une machine à tisser programmable à l'aide des cartes perforées





- ✚ 1821 - Charles Babbage invente la machine à différence, qui permet d'évaluer des fonctions.
- ✚ 1831 - Joseph Henry de Princeton invente le premier télégraphe.
- ✚ 1838 - Samuel Morse invente un code (plus tard appelé code Morse) qui utilise des numéros différents pour représenter les lettres de l'alphabet anglais et des chiffres.
- ✚ 1868 - Christopher Sholes invente la machine à écrire aux Etats-Unis en utilisant le clavier QWERTY. (Le clavier français est AZERTY).
- ✚ 1896 - Herman Hollerith fonda la Tabulating Machine Company, la société devient plus tard la célèbre société d'informatique IBM (International Business Machines).



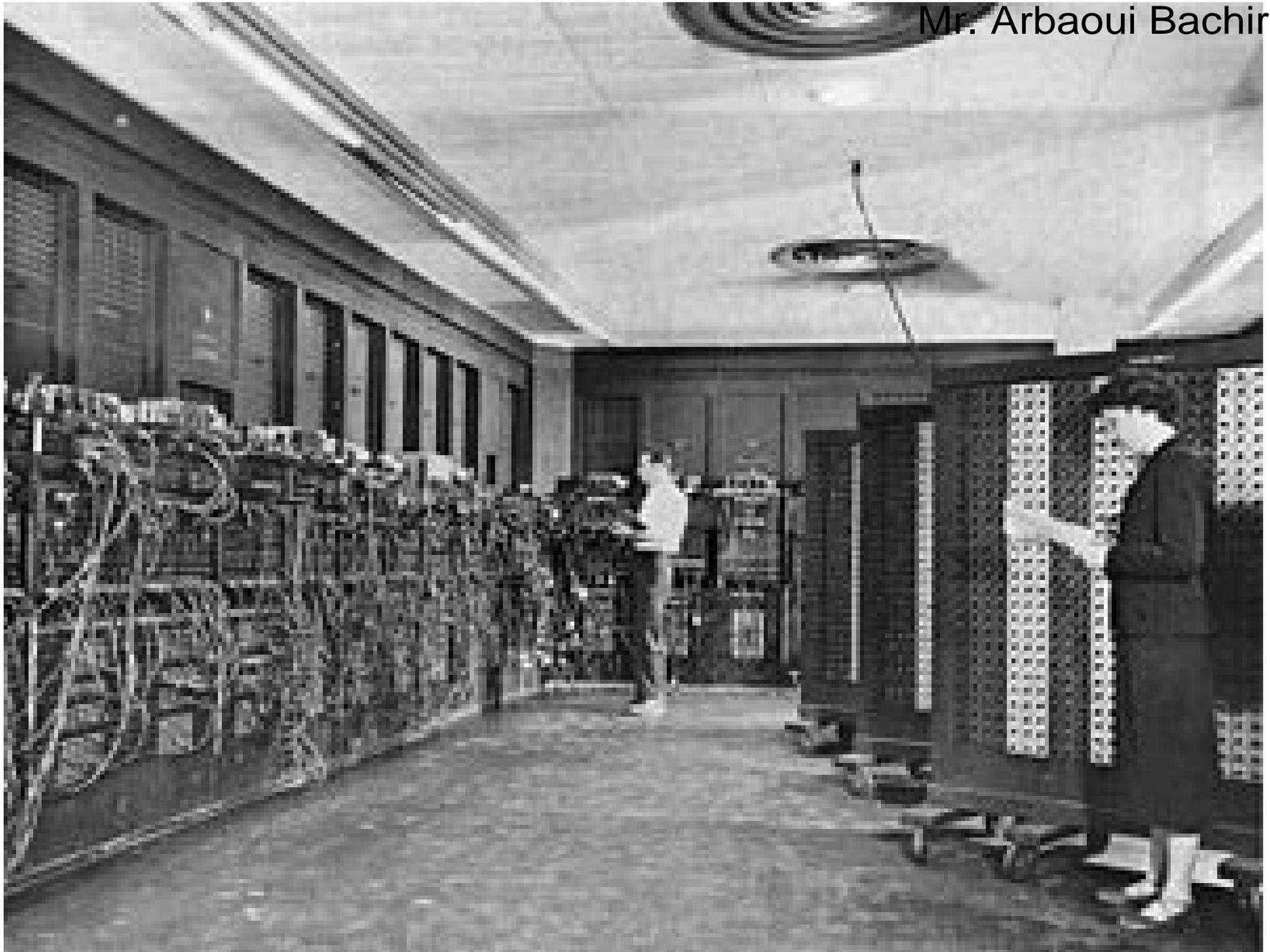
- ✚ 1937, Howard Aiken met au point un ordinateur programmable mesurant 17 m de long et 2.5 mètres de hauteur, permettant de calculer 5 fois plus vite que l'homme C'est le Mark I d'IBM.
- ✚ Il est constitué de 3300 engrenages, 1400 commutateurs reliés par 800 km de fil électrique.

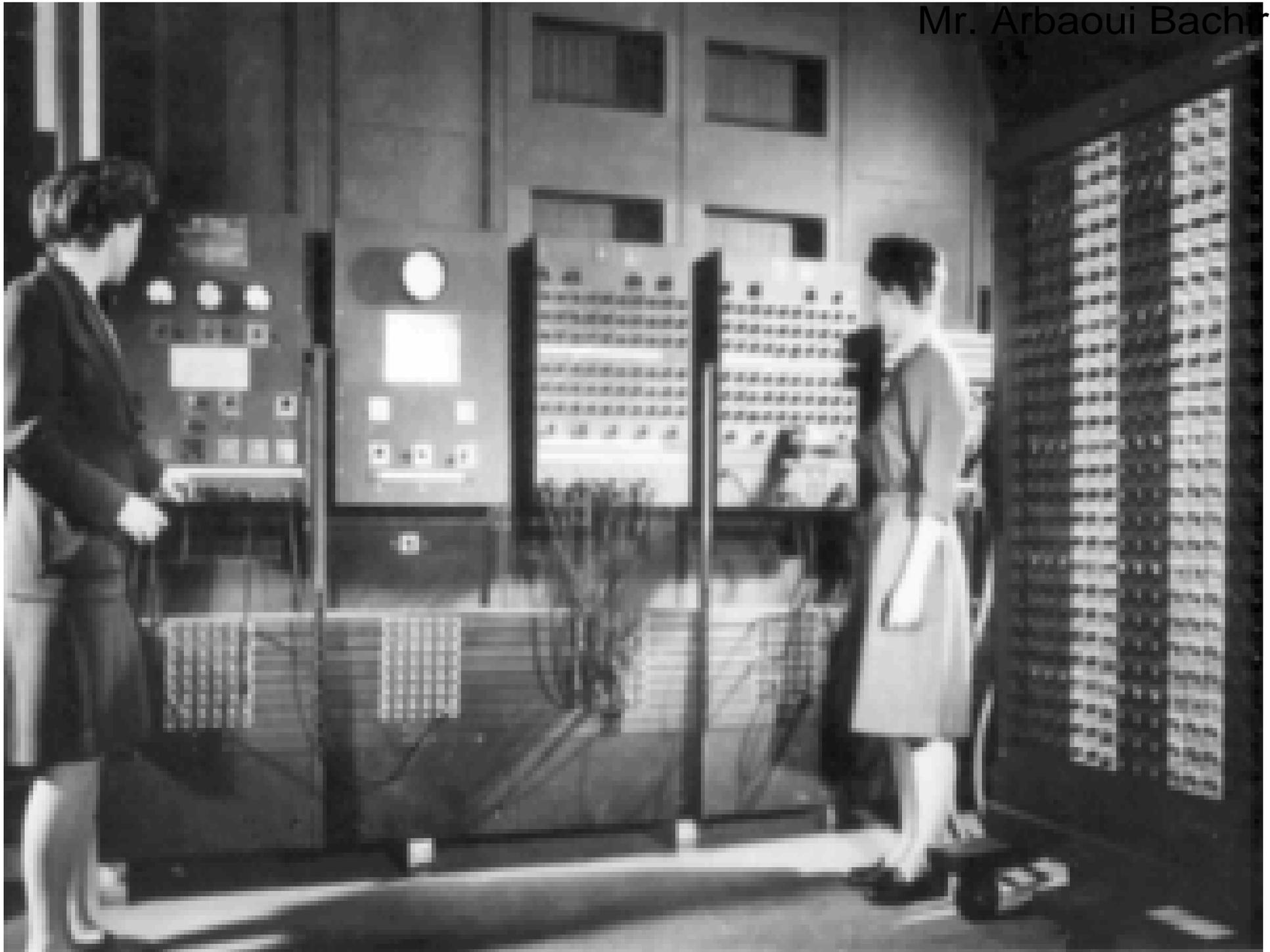




1943 - ENIAC

- Premier ordinateur ne comportant aucunes pièces mécaniques est créé grâce à J.Mauchly et J.Presper Eckert : l'ENIAC (Electronic Numerical Integrator And Computer). Il était le premier ordinateur purement électronique.
- L'ENIAC a été conçu et construit pour le calcul des tables de tir d'artillerie de l'armée américaine du Laboratoire de recherche en balistique et utilisé aussi pour la construction de la première bombe H.
- La taille physique de l'ENIAC était massive par rapport aux normes modernes des PC. Il contenait 17468 tubes à vide, 7200 diodes de cristal, 1500 relais, 70000 résistances et 10000 condensateurs. Il pesait 30 tonnes et occupe l'espace de 1500m². Sa consommation était de 150 kW de puissance.
- Prix: 500,000\$



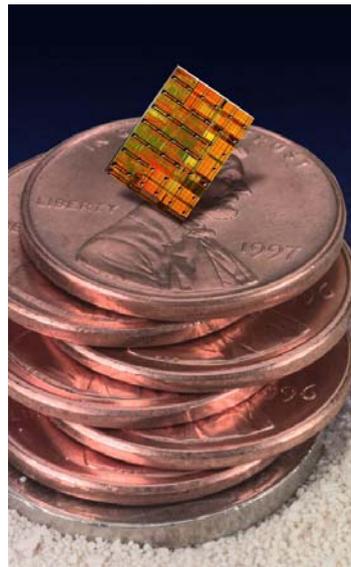




- La première erreur informatique est due à un insecte qui, attiré par la chaleur, était venu se loger dans les lampes et avait créé un court-circuit. Ainsi le terme anglais pour « insecte » étant « bug », le nom est resté pour désigner une erreur informatique.
- ✚ 1946 - Mise au point de l'EDVAC (Electronic Discrete Variable Computer) permettant de stocker les programmes en mémoire (1024 mots en mémoire centrale et 20000 mots en mémoire magnétique).
- ✚ 1948 - le transistor est créé par la firme Bell Labs (grâce aux ingénieurs John Bardeen, Walter Brattain et William Shockley). Il permet dans les années 50 de rendre les ordinateurs moins encombrants, moins gourmands en énergie électrique donc moins coûteux : c'est la révolution dans l'histoire de l'ordinateur !



- ✚ 1958 - Le circuit intégré est mis au point par Texas Instruments, il permet de réduire encore la taille et le coût des ordinateurs en intégrant sur un même circuit électronique plusieurs transistors sans utiliser de fil électrique.
- ✚ 1960 - IBM 7000 est le premier ordinateur à base de transistor.
- ✚ 1964 - IBM 360 fait son apparition, avec également l'arrivée remarquée du DEC PDP-8.





L'âge moderne

- ✚ 1971 – Le courrier électronique (Email) est inventé par Ray Tomlinson.
- ✚ 1975 - Bill Gates and Paul Allen forme un partenariat connu sous le nom de Microsoft et écrivent des logiciels informatiques.
- ✚ 1976 – La société Apple computers est créée.
- ✚ 1978 – Les lecteurs de disquettes 5 1/4” sont utilisées pour la première fois par Apple Computers.
- ✚ 1981 - Les premiers ordinateurs personnels (PC) utilisant le système d'exploitation MS-DOS sont vendus au public.
- ✚ 1982 – les CD de Musique et lecteurs de CD sont introduits au Japon.
- ✚ 1982 - TCP / IP, la langue commune de tous les ordinateurs sur Internet se développe.



- + 1984 - L'ordinateur Macintosh est lancé. C'est le premier ordinateur à utiliser la souris ainsi que des interfaces graphiques.
- + 1987 - IBM standardise le lecteur de disquette 3 1 / 2 " sur tous les ordinateurs.
- + 1988 - Le premier virus informatique, un ver, attaque et immobilise plus de 10% du trafic Internet.
- + 1991 - Le World Wide Web est inventé par le CERN.
- + 1995 - AOL, Prodigy, CompuServe offrent des services de connexion à internet pour les particuliers en utilisant les modem téléphoniques.
- + 1995 – Le film Toy Story est un long-métrage réalisé pour la première fois en utilisant des animations 100% par ordinateurs.
- + 1998 - Apple Computer introduit le iMac, qui élimine la nécessité d'une tour (le tout dans une machine)
- + 2005 - présent: Le wireless à la demande prend la scène



← Le premier PC d'IBM

Le Macintosh →





Constitution physique d'un ordinateur

- Un ordinateur est un ensemble de composants électroniques modulaires, c'est-à-dire des composants pouvant être remplacés par d'autres composants ayant éventuellement des caractéristiques différentes. Ces composants sont assemblés dans une tour, boîtier, appelée: Unité Centrale.

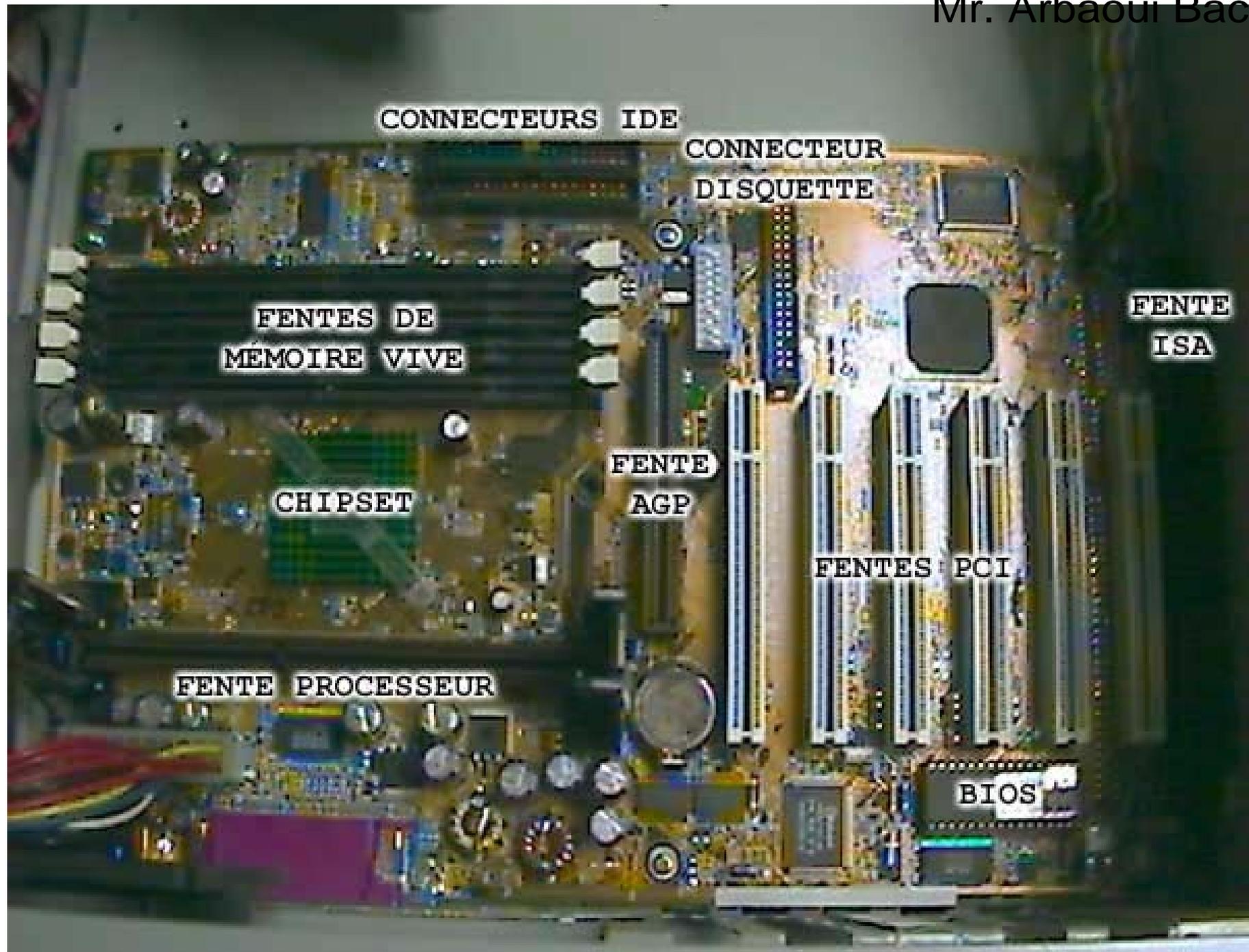




La carte mère

Elle supporte l'ensemble des éléments (microprocesseur, mémoire, cartes d'extension...).

C'est l'épine dorsale de l'ordinateur





La mémoire vive est la section dont le processeur se sert pour travailler les informations. Le processeur utilise beaucoup de mémoire vive, pour traiter l'information le plus rapidement possible. Si cette quantité est insuffisante, pour cause de données trop importantes (une image, une vidéo, une encyclopédie ou une base de données), le processeur demande au disque dur de stocker le surplus de données.





- **Le disque dur** est conçu pour mémoriser des informations au moment considéré : il les transfère jusqu'à 100 fois moins vite que la mémoire vive. Pour que votre PC aille vite, il lui faut suffisamment de mémoire vive (16 Mo minimum pour Windows 95 et 32 Mo pour Windows 98/Me, 128 pour XP).

Le disque dur est l'outil indispensable de stockage d'informations dynamiques : il est capable de stocker des quantités d'informations phénoménales (1 Go = 1 million de pages dactylographiées A4 contenant 1000 caractères chacune), et de les modifier rapidement. Le disque dur stocke aussi bien le programme utilisé par le processeur pour effectuer certaines tâches (traitement de texte, logiciel de dessin, tableur, logiciel de montage vidéo, etc.) et le système d'exploitation du PC (D.O.S., Windows 95, Windows 98, Windows Me, etc). **Le disque dur est d'une importance fondamentale, et tout dysfonctionnement à ce niveau perturbe très gravement l'utilisation du P.C.**





L'écran sert à l'affichage des images et de toutes les informations que l'ordinateur doit vous montrer et sur lesquelles vous intervenez.

Le clavier permet d'entrer des commandes ou de taper du texte.

La souris sert à pointer, cliquer sur des icônes pour lancer des actions ou des programmes, jouer et même à dessiner. L'utilisation de la souris demande un certain temps d'apprentissage aux novices, mais le confort qu'offre une souris de bonne qualité est remarquable.

La ventilation de l'alimentation est à contrôler régulièrement car un dysfonctionnement et vous grillez votre ordinateur dans le pire des cas.





LE MICROPROCESSEUR

C'est le " cerveau "de l'ordinateur. Il est fixé sur la carte mère. Actuellement les processeurs sont des **32 bits**, ce qui signifie qu'ils sont capables de traiter 4 caractères à la fois (un caractère = un octet = 8bits)

Les principaux fabricants sont Intel (Pentium II, Pentium III, Pentium IV, Céléron, ...) et AMD (Duron, Thunderbird, Atlon, ...).

Le processeur (CPU : *Central Processing Unit*) est un circuit électronique cadencé au rythme d'une horloge interne, c'est-à-dire un élément qui envoie des impulsions ou battements (que l'on appelle *top*). A chaque *top* d'horloge les éléments de l'ordinateur accomplissent une action. La vitesse de cette horloge (le nombre de battements par secondes) s'exprime en Mégahertz, ainsi un ordinateur à 500Mhz a une horloge envoyant 500 000 000 de battements par seconde (un cristal de quartz soumis à un courant électrique permet d'envoyer des impulsions à une fréquence précise).



Le microprocesseur contient plusieurs transistors:

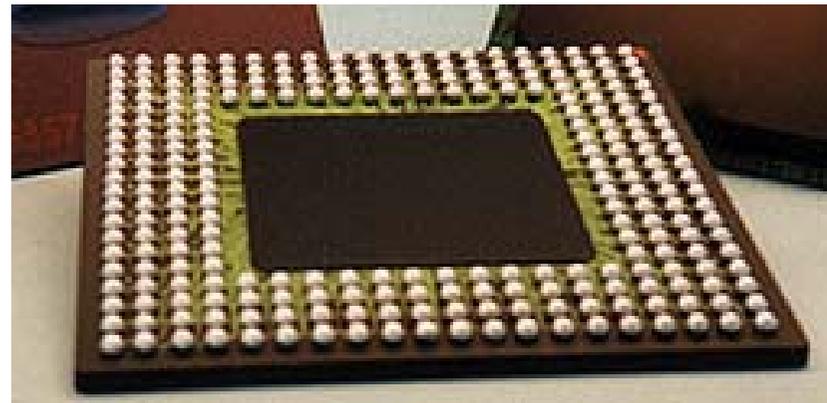
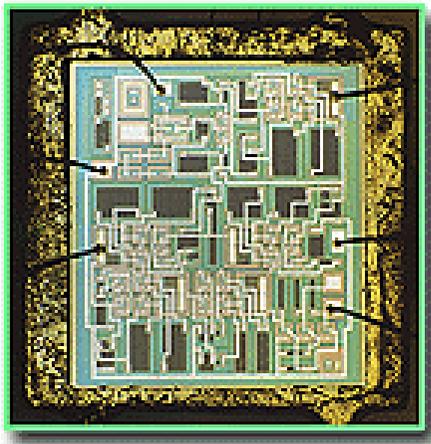
(ENIAC): 19,500 tubes à vide et relais

Intel 8088 processor (1er PC): 29,000 transistors

Intel Pentium II processeur: 7 millions transistors

Intel Pentium III processeur: 28 millions transistors

Intel Pentium 4 processeur: 42 millions transistors





LA GRANDE QUESTION:

QUI VA GERER TOUS CE MATERIEL DE L'UNITE CENTRALE A L'AFFICHAGE A L'ECRAN PASSANT PAR LE CLAVIER?

LE SYSTEME D'EXPLOITATION



Cours N°2

Systemes d'Exploitation

Mr. Arbaoui Bachir



Introduction

Pour qu'un ordinateur soit capable de faire fonctionner un programme informatique, la machine doit être en mesure d'effectuer un certain nombre d'opérations préparatoires afin d'assurer les échanges entre le processeur, la mémoire et les ressources physiques (périphériques).



Description d'un système d'exploitation

Un système d'exploitation (*noté **SE** ou **OS**, abréviation du terme anglais **Operating System***), d'un ordinateur ou d'une installation informatique est un ensemble de programmes qui remplissent deux grandes fonctions :

- ✚ Gérer les ressources de l'installation matérielle
- ✚ Assurer un ensemble de services en présentant aux utilisateurs une interface mieux adaptée à leurs besoins que celle de la machine physique



Rôles du système d'exploitation

Les rôles du système d'exploitation sont divers :

- *Gestion du processeur*
- *Gestion de la mémoire vive*
- *Gestion des entrées/sorties*
- *Gestion de l'exécution des applications*
- *Gestion des droits*
- *Gestion des fichiers*
- *Gestion des informations*



Historique des systèmes d'exploitation

- ✚ **1^{ère} Génération (1945 - 1955) : les tubes à vide et les cartes perforées**
 - Systèmes d'exploitation inexistants

- ✚ **2^{ème} Génération (1955 - 1965) : les transistors et le traitement par lots**
 - Les ordinateurs sont devenus plus fiables grâce à l'apparition des transistors
 - Commercialisation des ordinateurs à des utilisateurs (grandes compagnies, université ou administrations...)
 - Les utilisations principales étaient le calcul scientifique et l'ingénierie (équations différentielles)
 - Apparition des langages FORTRAN et assembleur
 - Apparition des systèmes d'exploitation (SE) : FMS (Fortran Monitor System) et IBSYS (IBM 7094)



Historique des systèmes d'exploitation

- ✚ **3^{ème} Génération (1965 - 1980) : les circuits intégrés et la multiprogrammation**
 - Amélioration des coûts et des performances (circuits intégrés)
 - Apparition d'une famille d'ordinateurs compatibles entre eux avec une seule architecture et un même jeu d'instructions. Des ordinateurs uniques pour les calculs scientifiques et commerciaux
 - Apparition aussi du spoule (spool, Simultaneous Peripheral Operation On Line) pour le transfert des travaux des cartes vers le disque
 - Apparition de la multiprogrammation (partitionnement de la mémoire pour des tâches différentes). Mais, un système d'exploitation énorme et très complexe pour satisfaire tous les besoins (plusieurs millions de lignes d'assembleur)



Historique des systèmes d'exploitation

- ✚ **3^{ème} Génération (1965 - 1980) : les circuits intégrés et la multiprogrammation**
 - Apparition du partage de temps, une variante de la multiprogrammation (chaque utilisateur possède un terminal en ligne) ; naissance du système MULTICS (MULTIplexed Information and Computing Service) pour ordinateur central
 - Apparition d'une version simplifiée (mono-utilisateur) de MULTICS appelée UNICS (Uniplexed Information and Computer Service) ; ce nom allait devenir UNIX1. Par la suite il y a réécriture de UNIX en langage C ; ce système d'exploitation a été le plus porté sur toutes sortes de machine



Historique des systèmes d'exploitation

- ✚ **4^{ème} Génération (1980 - 1990) : les ordinateurs personnels**
 - Développement des circuits LSI (Large Scale Integration) contenant des centaines de transistors au cm^2
 - Il existe deux principaux systèmes d'exploitation : MS-DOS (Microsoft Inc.) et UNIX. MS-DOS intègre petit à petit des concepts riches d'UNIX et de MULTICS
 - Apparition de réseaux d'ordinateurs individuels qui fonctionnent sous des systèmes d'exploitation en réseau ou des systèmes d'exploitation distribués



Historique des systèmes d'exploitation

- **5^{ème} Génération (1990 - ????)** : les ordinateurs personnels portables et de poche
 - Apparition des PIC (Personal Intelligent Communicator) et des PDA (Personal Digital Assistant), grâce à l'intégration des composants et l'arrivée des systèmes d'exploitation de type «micro-noyau» utiles pour les systèmes de gestion des informations
 - Les systèmes d'exploitation de type « micro-noyau » sont modulaires (un module par fonction) ; ils peuvent être réalisés avec plus ou moins de modules et donc adaptables à des très petites machines (PDA et PIC)



Composantes du système d'exploitation

Le système d'exploitation est composé d'un ensemble de logiciels permettant de gérer les interactions avec le matériel. On distingue généralement les éléments suivants :

- **Le noyau** représente les fonctions fondamentales du système d'exploitation (la gestion de la mémoire, des processus, des fichiers, des entrées-sorties et des fonctionnalités de communication)



Composantes du système d'exploitation

- ***L'interpréteur de commande*** permette la communication avec le système d'exploitation par l'intermédiaire d'un langage de commandes afin de permettre à l'utilisateur de piloter les périphériques en ignorant tout des caractéristiques du matériel qu'il utilise, de la gestion des adresses physiques, etc.
- ***Le système de fichiers*** permette d'enregistrer les fichiers dans une arborescence



Qualités d'un système d'exploitation

- **Fiabilité** : Limiter les conséquences des défaillances matérielles ou des erreurs des utilisateurs. En cas de panne, éviter les pertes d'information ou leur incohérence
- **Efficacité** : Utiliser au mieux les ressources et possibilités matérielles
- **Facilité d'emploi** : Offrir un langage de commande (dialogue usager/système) et des diagnostics d'erreurs (système/usager) clairs et précis



Qualités d'un système d'exploitation

- **Adaptabilité** : Permettre des modifications matérielles et logicielles les plus simples possibles, à l'aide d'outils spécialisés
- **Mesurabilité** : Enregistrer la comptabilité des ressources utilisées par les usagers, mesurer les paramètres de fonctionnement et de charge



Différentes classes de systèmes d'exploitation

- **Systemes multitâches** : Un système d'exploitation est dit «multi-tâche» lorsque plusieurs «tâches» (également appelées processus) peuvent être exécutées simultanément
- **Systemes multi-processeurs** : Le multiprocessing est une technique consistant à faire fonctionner plusieurs processeurs en parallèle afin d'obtenir une puissance de calcul plus importante que celle obtenue avec un processeur



Différentes classes de systèmes d'exploitation

- **Systemes embarqués** : Les systèmes embarqués sont des systèmes d'exploitation prévus pour fonctionner sur des machines de petite taille ou des appareils électroniques autonomes (sondes spatiales, robot, ordinateur de bord de véhicule, etc.), possédant une autonomie réduite
- **Systemes temps réel** : Les systèmes temps réel, essentiellement utilisés dans l'industrie, sont des systèmes dont l'objectif est de fonctionner dans un environnement contraint temporellement



Quelques types de systèmes d'exploitation

Caractéristiques	DOS	MAC OS (Apple)	WINDOWS	WIN NT	UNIX (AIX, LINUX, SOLARIS...)	GCOS (BULL)
Ouvert					X	
Monotâche	X					
Multitache		X	X	X	X	X
Mono-utilisateur	X	X	X	X		
Multi-utilisateur					X	X
Multiprocesseurs				X	X	X



Notion de Fichier

✚ Définitions

- Un fichier informatique est une unité informationnelle physiquement stockée sur un support de mémoire de masse permanent (disque dur par exemple)
- Le fichier est donc enregistré sur le disque dur sous la forme "nom_du_fichier.ext"
".ext" représente l'extension (c'est un moyen de reconnaître le type de programme avec lequel ce fichier peut être ouvert)
- Un fichier peut être une suite d'informations binaires, c'est-à-dire une suite de 0 et de 1. Ce fichier peut être stocké pour garder une trace d'informations
- Un fichier a généralement comme attributs : un nom ; un chemin d'accès ; une taille mesurée en octets ; une date de création et une de dernière modification



Note : L'octet est une unité de mesure en informatique mesurant la quantité de données. Un octet est lui-même composé de 8 bits, soit 8 chiffres binaires. Le byte, qui est un ensemble de bits adjacents, a presque toujours une taille d'un octet

Bits \neq Byte

Multiples d'octets Usage traditionnel	
Symbole	Valeur
Ko	2^{10} octets = 1024 octets
Mo	2^{20} octets = 1024 Ko
Go	2^{30} octets = 1024 Mo



Notion de Fichier

✚ Règles à suivre pour nommer un fichier

- Ne pas utiliser de caractères accentués (Ex : "?", ".", ";" ...)
- Ne pas utiliser d'espaces blancs: utiliser à la place « _ » ou des majuscules à l'intérieur du nom propre (MonFichier.txt ou mon_fichier.txt)
- Eviter d'utiliser le point à l'intérieur du nom propre de fichier ainsi que /
- Garder le nom de fichier aussi court que possible



Notion de Fichier

✚ Quelques extensions de fichiers

- *Contenant du texte*

.HTML .HTM .SHTM .SHTML .PHP .ASP .RTF .TXT .DOC
.PDF .PS .EPS .TEX .DVI .XLS

- *Compression, archivage et encodage*

.ZIP .BIN .EXE .HGX .SIT .UUE .RAR

- *Les extensions souvent recompressées*

.GZ .TAR .Z

- *Multimédia*

.PPT .PPS .DIR .SWF



Notion de Fichier

+ Quelques extensions de fichiers

■ *Images*

.PS .EPS .BMP .GIF .JPEG .JPG .TIFF .TIF

■ *Audio*

.AIFF .AIF .MID .MP3 .RA .RAM .WAV

■ *Vidéo*

.AVI .MOV .QT .MPEG .VDO .VIV



Notion de Fichier

✚ Infection par un virus

Lorsqu'un fichier est infecté par un virus, ce dernier y ajoute des lignes de code. Ainsi, l'information concernant la taille du fichier située dans l'en-tête ne correspondra plus (à moins que le virus ne soit programmé de manière à modifier l'en-tête), il pourra donc être repéré

✚ Organisation des fichiers en répertoires

Un disque dur peut contenir, suivant sa taille, plusieurs milliers de fichiers. Cependant, plus leur nombre est élevé plus il est difficile de les gérer, il devient alors nécessaire de les stocker dans des répertoires



Notion de Répertoire

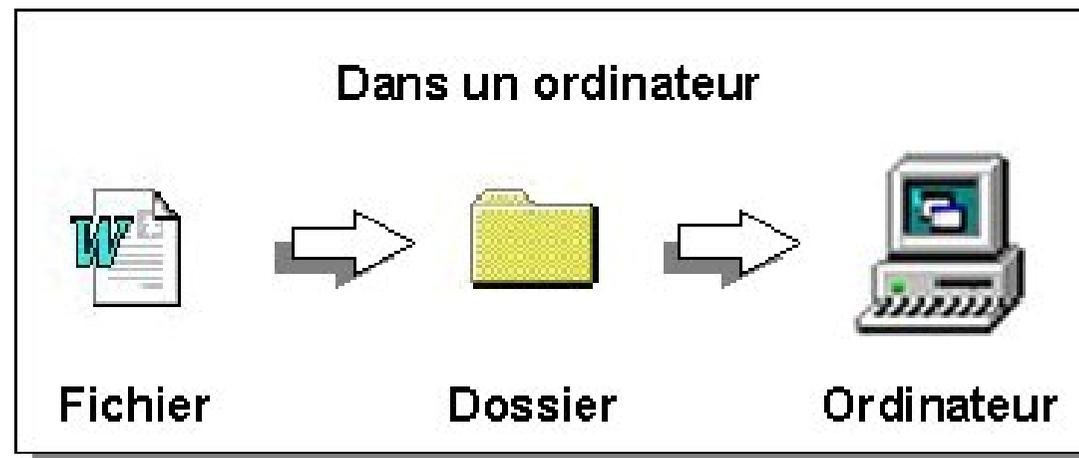
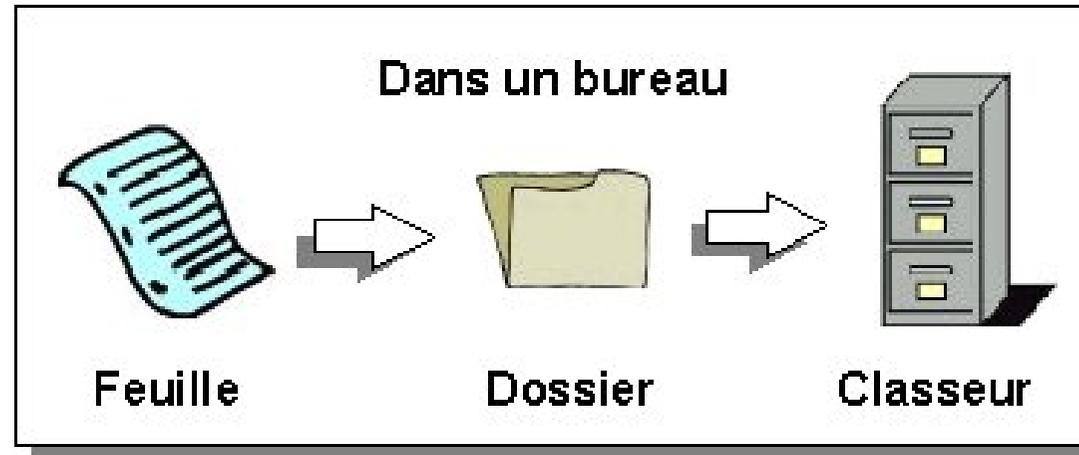
✚ Définition

Un répertoire (appelé également dossier ou folder en anglais) est un objet informatique pouvant contenir des fichiers. Un répertoire peut en effet contenir :

- *Des fichiers*
- *D'autres répertoires (Sous répertoire)*

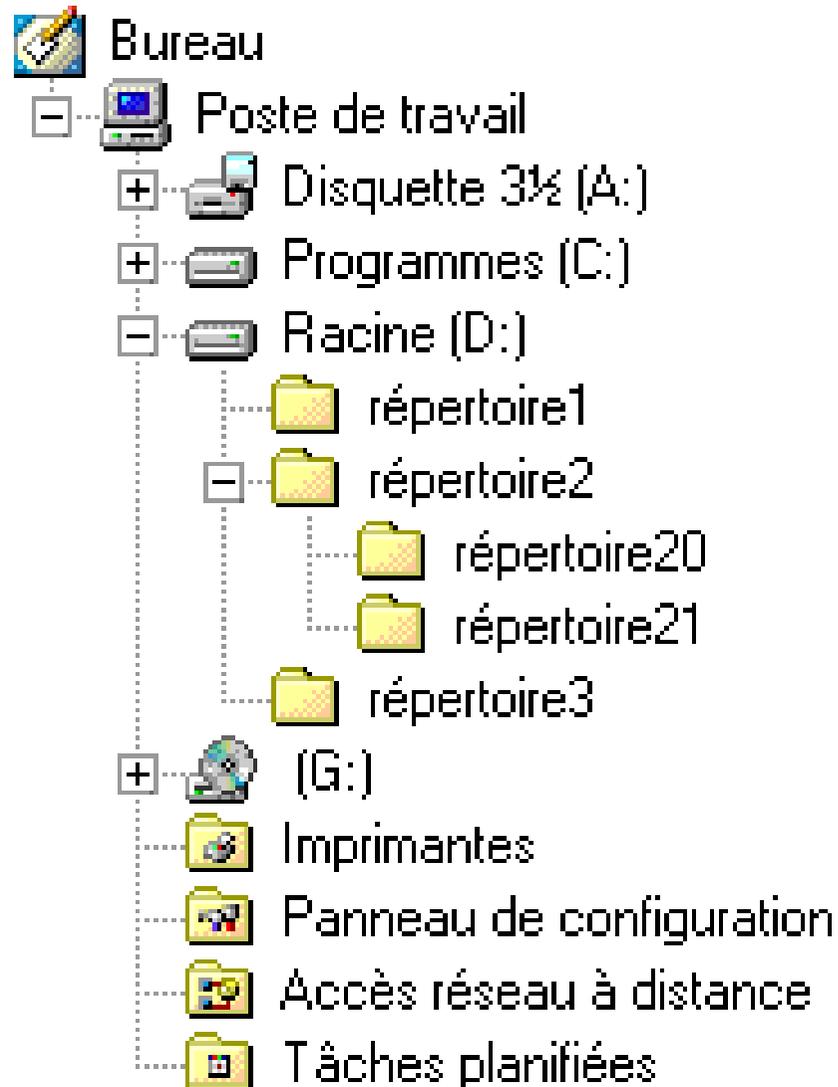


Notion de Répertoire





Notion de Répertoire





Notion de Répertoire

■ Notion de chemin

On appelle «chemin» (en anglais path) la succession des répertoires en partant de la racine pour atteindre un fichier.

- ❑ Sous les systèmes MS DOS et Windows un chemin est de la forme

`x:\repertoire1\repertoire2\`

- ❑ Sous un système Unix un chemin est de la forme

`/repertoire1/repertoire2/`



Microsoft DOS

✚ Présentation du système

- Le DOS est un système d'exploitation monotâche et mono-utilisateur, développé initialement par la société Microsoft pour gérer les PC d'IBM
- Le DOS contrôle les activités de l'ordinateur. Il gère des opérations telles que la circulation, l'affichage et l'entrée de données entre les divers éléments constitutifs du système.



Microsoft DOS

■ Rôle du DOS

- Le rôle du DOS est d'interpréter les commandes saisies au clavier par l'utilisateur
- Ces commandes permettent d'effectuer les tâches suivantes :
 - ✓ *La gestion des fichiers et des répertoires*
 - ✓ *La mise à jour des disques*
 - ✓ *La configuration du matériel*
 - ✓ *L'optimisation de la mémoire*
 - ✓ *L'exécution des programmes*



Microsoft DOS

■ Utilisation des commandes

- Ces commandes sont tapées à l'invite, c'est-à-dire une lettre d'unité suivi d'un antislash, ce qui donne A:\ ou C:\ par exemple. `C:\>`
- Pour exécuter une commande il suffit de taper la commande puis d'appuyer sur ENTREE



Historique – Evolution du MS DOS

- **MS-DOS 1.00** : *Août 1981*, sort avec le tout premier IBM PC. Ne connaît que les disquettes. Pas de répertoires, et pas plus de 64 fichiers par disquette
- **MS-DOS 2.00** : *Mars 1983*, première version à supporter un disque dur et des répertoires
- **MS-DOS 3.00** : *Août 1984*, supporte les disquettes de 1,2 MB et les disques durs de plus de 10 MB
- **MS-DOS 4.00** : *Juin 1988*, il occupe 110 KiB de RAM, offre une interface graphique (le Shell) et gère les disques durs supérieurs à 32 MB grâce à la FAT16
- **MS-DOS 5.00** : *Juin 1991*, version gérant la mémoire haute, et qui permet de faire des menus de démarrage ; supporte jusqu'à 8 disques dur, support des disquettes de 2.8 MiB



Historique – Evolution du MS DOS

- **MS-DOS 6.00** : *Mars 1993*, version intégrant un anti-virus, un outil de vérification de système de fichiers, ScanDisk, et un logiciel de compression, DoubleSpace
- **MS-DOS 6.22** : *Juin 1994*, DoubleSpace est remplacé par DriveSpace, logiciel compatible sans le problème de brevet. C'est la dernière version autonome de MS-DOS
- **MS-DOS 7** : *Août 1995*, le DOS de Windows 95
- **MS-DOS 7.1** : *Août 1996*, le DOS de Windows 95B et C et de Windows 98 et 98SE
- **MS-DOS 8** : *2000*, dernière version de MS-DOS. Il est intégré à Windows ME



Principales commandes MS DOS

Commandes DOS relatives aux répertoires

- **MD** ou **MKDIR** : Créer un répertoire

Syntaxe : **md <nom_repertoire>**

Ex: `c:\>md rep1`

`c:\>md a:\rep2`

- **CD** : Changer de répertoire

Syntaxe : **cd<chemin>** , **cd..**

Ex: `c:\>cd rep1`

`c:\rep1>cd..` (Permet de sortir du répertoire courant)

- **RD** : Supprimer un répertoire

Syntaxe : **rd<chemin>** [/s(pour supprimer une arborescence pleine)]

Ex: `c:\>rd rep1 /s`

`c:\>rd a:\rep2`



Principales commandes MS DOS

■ Commandes DOS relatives aux répertoires

- **DELTREE** : Supprimer un répertoire vide ou non vide

Syntaxe **deltree**<chemin>

```
Ex: c:\>deltree rep1  
c:\>deltree a:\rep2
```

Rem: cette commande n'existe plus sur les versions Windows 2000 et XP

- **DIR** : Afficher le contenu d'un répertoire

Syntaxe : **dir** [<chemin>] [/P(liste)] [/W(liste horizontal)]
[/O(x)(liste par ordre)] [/A(x)(liste selon les attributs des
fichiers)] [/S(liste tous les répertoires et sous répertoires avec
leurs fichiers)] [/B(liste avec le format abrégé)] [/L(liste en
minuscule)]

```
Ex: c:\>dir rep1/P/OD  
c:\>dir /W/AH
```



Principales commandes MS DOS

■ Commandes DOS relatives aux répertoires

- **XCOPY** : Copier un répertoire

Syntaxe : **xcopy**<source>[<destination>] [/e(copier tous le répertoires y compris les sous répertoires avec leurs fichiers)]
[/s(copier tous le répertoires y compris les sous répertoires avec leurs fichiers sauf les vides)]

Ex:

```
c:\>xcopy rep1 a:\ /e
```

- **TREE** : Représente graphiquement l'arborescence d'un lecteur ou d'un chemin

Syntaxe : **tree**<chemin> [/F(Affiche le nom des fichiers de tous les répertoires)]

Ex:

```
c:\>tree rep1  
C:\rep1>tree/F
```



Principales commandes MS DOS

■ Commandes DOS relatives aux fichiers

- **EDIT** : Allez à l'éditeur de texte DOS ou Créer un fichier avec l'éditeur

Syntaxe : **edit <nom_fichier>**

Ex: `c:\rep1>edit essai.txt`

- **DEL** : Supprimer un fichier

Syntaxe : **del [<chemin>]<nom_fichier>**

Ex: `c:\rep1>del essai.txt`

`c:\>del c:\rep1\essai.txt`

- **REN** ou **RENAME** : Renommer un fichier

Syntaxe : **ren <ancien_nom> <nouveau_nom>**

Ex:

`c:\rep1>ren essai.txt texte.txt`

`c:\>ren c:\rep1\essai.txt texte.txt`



Principales commandes MS DOS

■ Commandes DOS relatives aux fichiers

- **TYPE** : Afficher le contenu d'un fichier

Syntaxe : **type <nom_fichier>**

Ex: `c:\rep1>type essai.txt`

- **COPY** : Copier un fichier

Syntaxe **copy <source> [<destination>] [/v(vérifie que les informations sont copiées correctement)]**

Ex: `c:\rep1>copy essai.txt a:`

`c:\>copy c:\rep1\essai.txt a :`

- **MOVE** : Déplacer un fichier ou renommer un répertoire

Syntaxe : **move <source> [<destination>]**

Ex: `c:\>move c:\rep1\essai.txt c:\rep2\`

`C:\>move rep1 rep5`



Principales commandes MS DOS

✚ Commandes DOS relatives aux fichiers

- **ATTRIB** : Changer les attributs d'un fichier
Syntaxe : **attrib (-/+r** (fichier en lecture seule), **-/+a** (fichier archive), **-/+s** (fichier système), **-/+h** (fichier caché))
<nom_fichier>

Ex: `c:\rep1>attrib +h essai.txt`

✚ Autres commandes

- Formater une unité : **format <unité>**
Ex: `c:\>format a:`
- Copier le contenu d'un disque vers un autre :
diskcopy<unité1> <unité2>

Ex: `c:\>diskcopy a: a: (dans ce cas de a vers a)`



Principales commandes MS DOS

✚ Autres commandes

- Efface l'écran : **cls**
- Donne le numéro de version : **ver**
- Affiche le nom et le numéro de série du volume : **vol**
- Vérifie et corrige les erreurs sur le disque X :
CHKDSK <unite>[/f]
Ex: `c:\>chkdsk a :/f`
- Affiche ou définit l'heure de l'horloge interne du système :
time
- Aide sur la commande demandée : **(Commande)/?**
Ex: `c:\>attrib/?`



Principales commandes MS DOS

✚ La sélection de fichiers avec (*)

La commande générique (*) permet de réaliser des sélections de fichiers en remplaçant n'importe quelle chaîne de caractères. Elle s'utilise avec quelques commandes MS DOS

Exemple :

.	tous les fichiers
*.txt	tous les fichiers textes
b*.exe	tous les fichiers exécutables commençants par la lettre b



Microsoft Windows

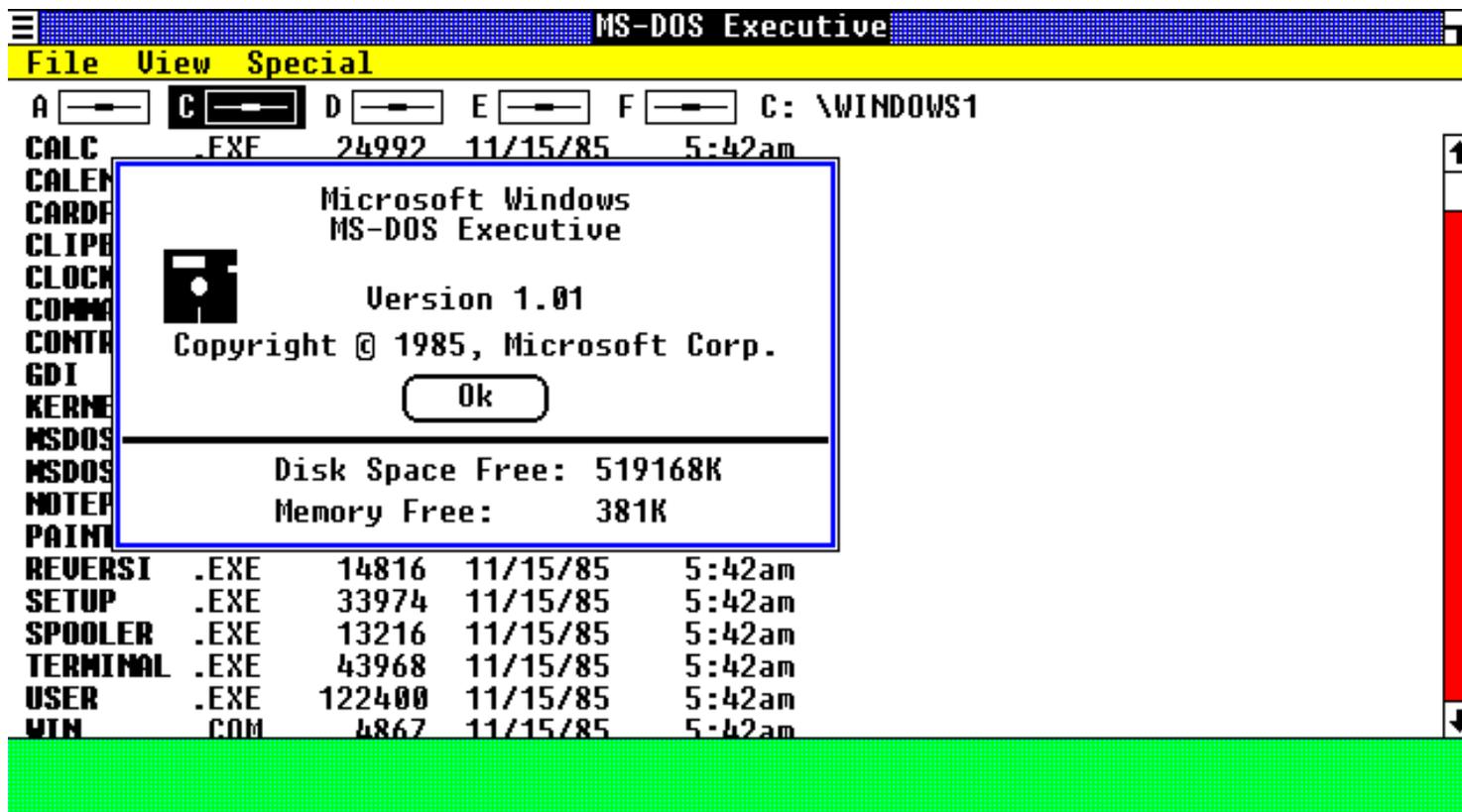
■ Présentation du système

- Gamme de systèmes d'exploitation développés par Microsoft et destinés aux ordinateurs compatibles PC
- Il permet une gestion cohérente, normalisée, à l'aide de symboles, menus et champs de dialogue graphiques que l'on active généralement par un clic de la souris
- Il n'est plus nécessaire de saisir les commandes manuellement, comme c'était le cas avec MS-DOS
- Le nom "Windows" provient du fait que l'on utilise des fenêtres pour représenter la surface de travail sur laquelle on exploite les programmes d'application et les documents



Historique – Evolution du Windows

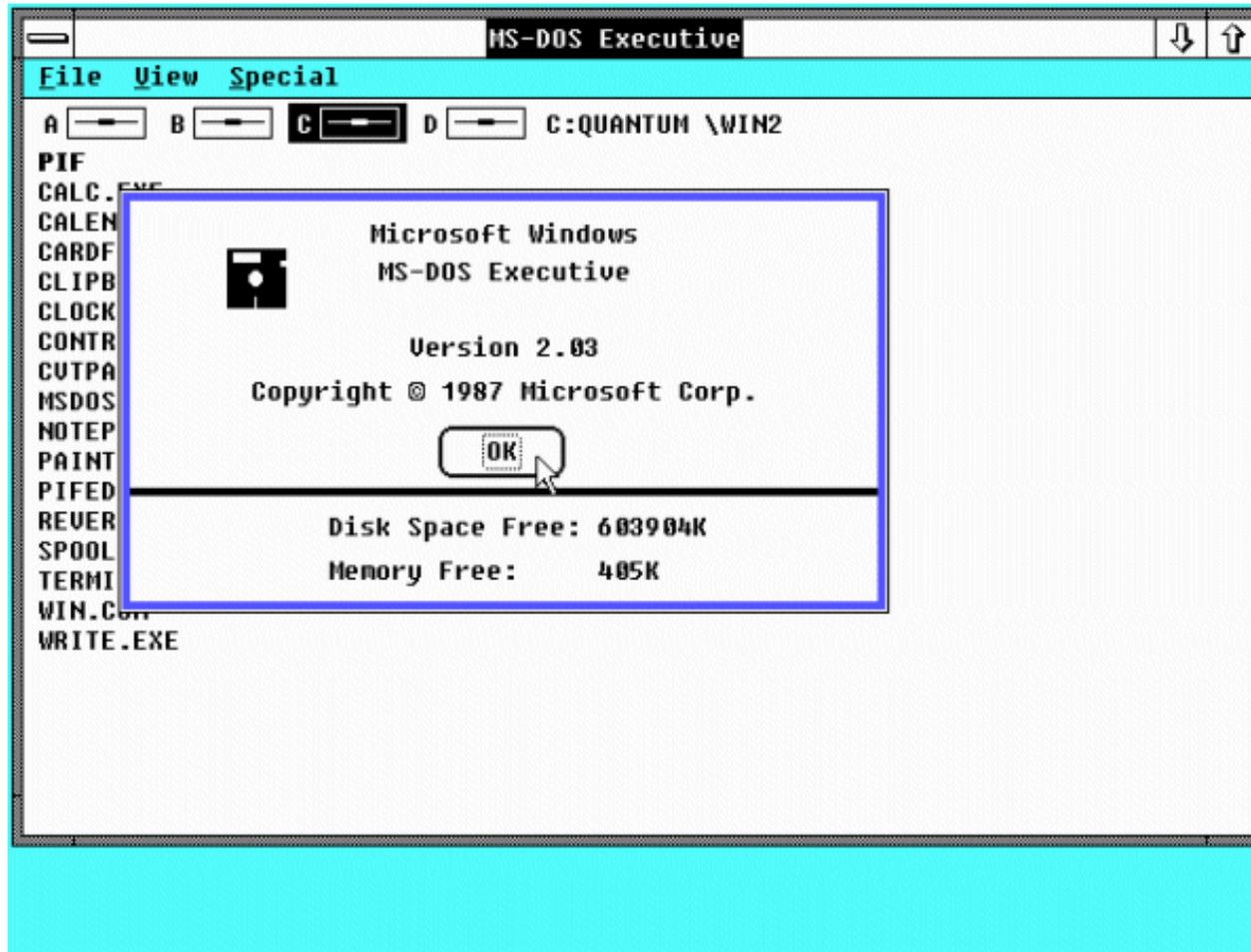
- **Windows 1.0** : *Novembre 1985*, cette version reprenait le concept d'interface graphique à fenêtres, déjà utilisé par la firme Apple dans ses machines Lisa et Macintosh





Historique – Evolution du Windows

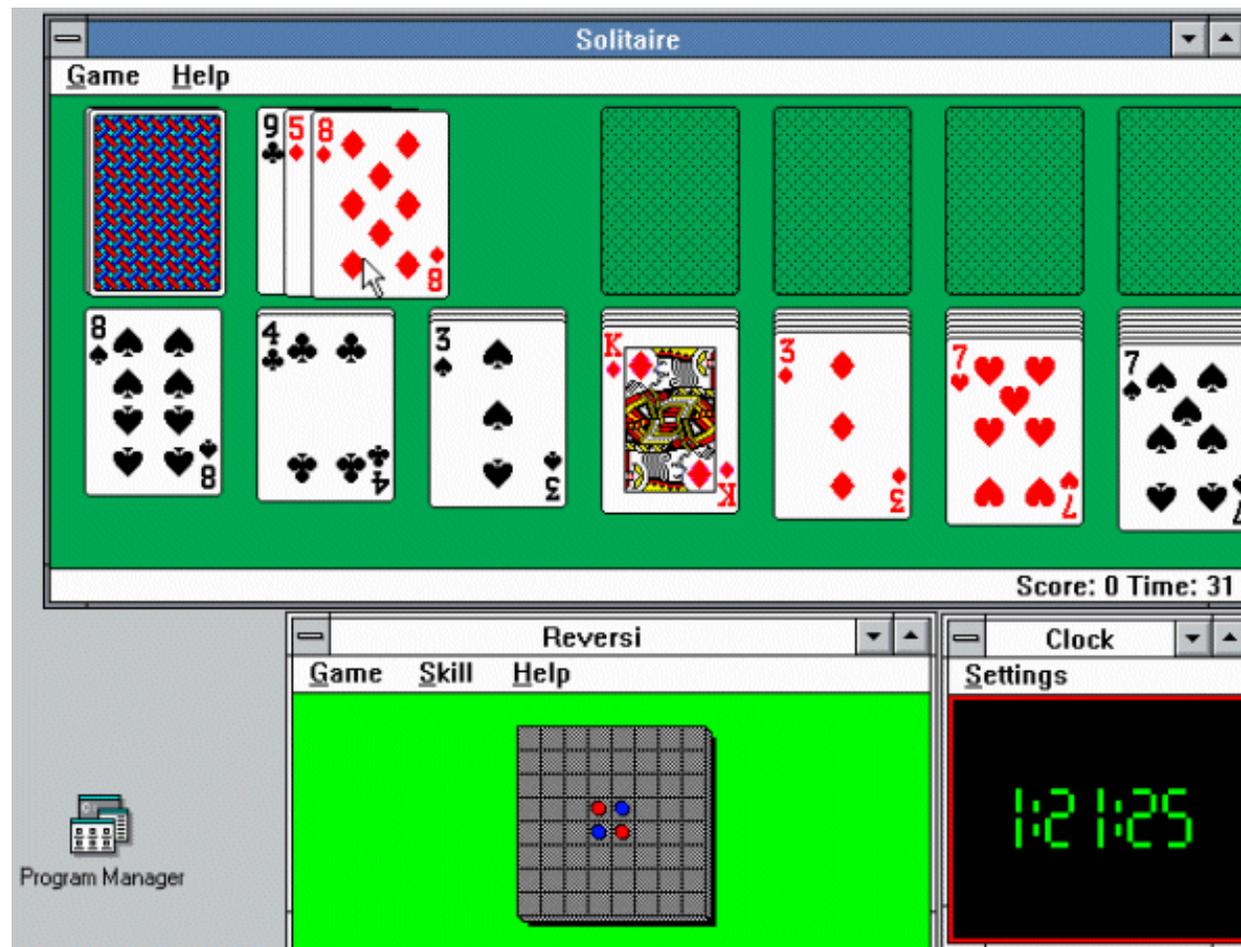
- **Windows 2** : *Décembre 1987*, apparition des Fenêtres superposables





Historique – Evolution du Windows

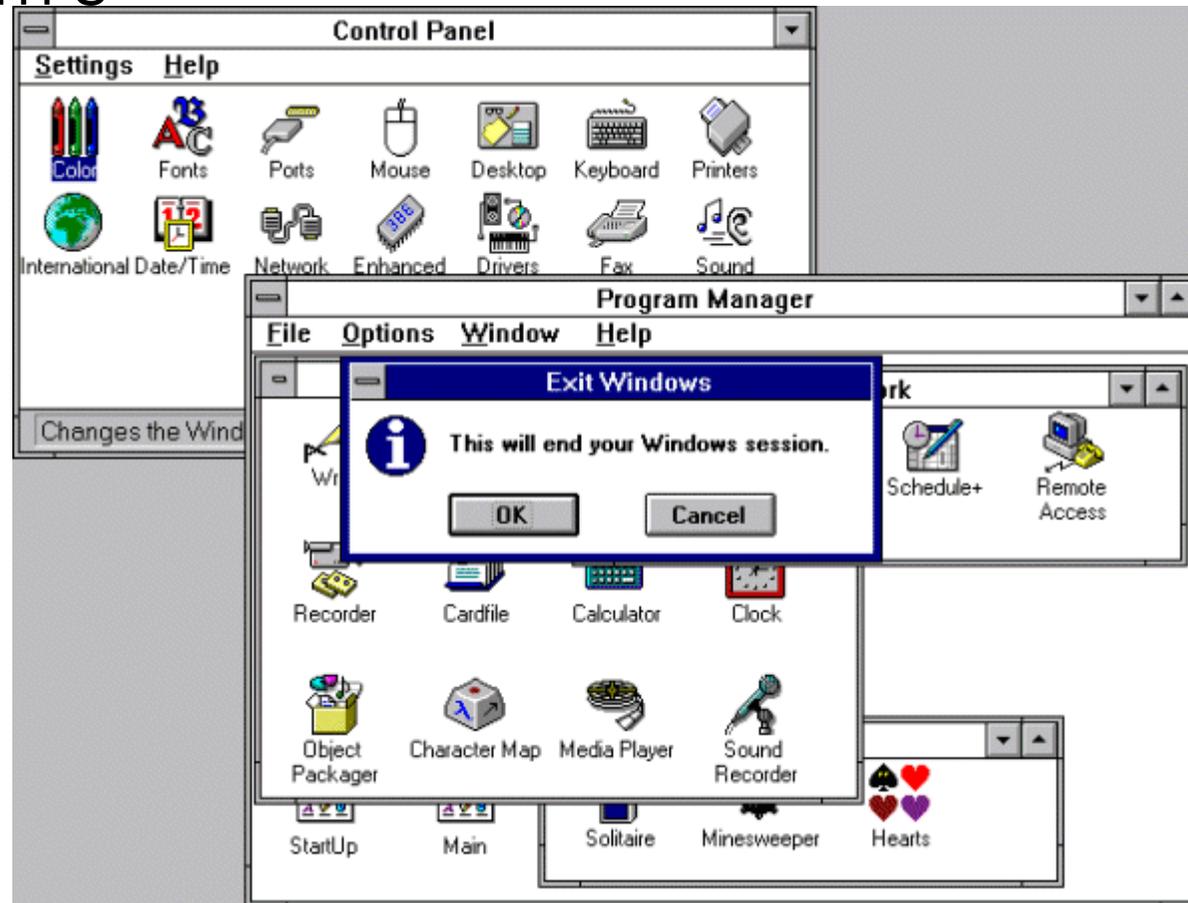
- **Windows 3.0** : Mai 1990, système utilisant 16 couleurs ainsi qu'un gestionnaire de programme





Historique – Evolution du Windows

- **Windows NT 3.1** : *Juillet 1993*, La première version, après de longues phases de bétas version : le principal y est : architecture 32 bits, multitâche préemptif, support multi processeurs, système de fichiers NTFS





Historique – Evolution du Windows

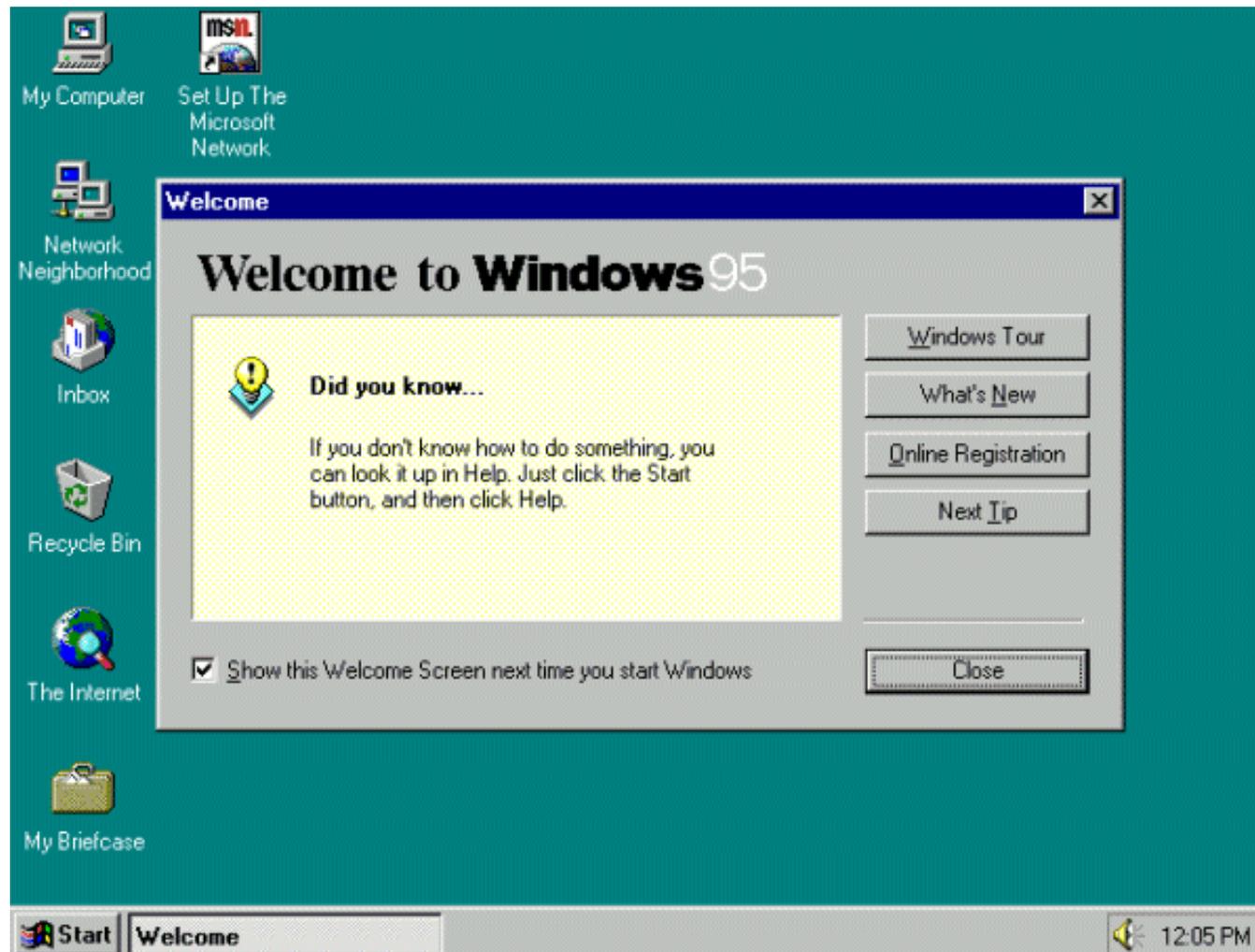
■ Windows 3.11 : Novembre 1993





Historique – Evolution du Windows

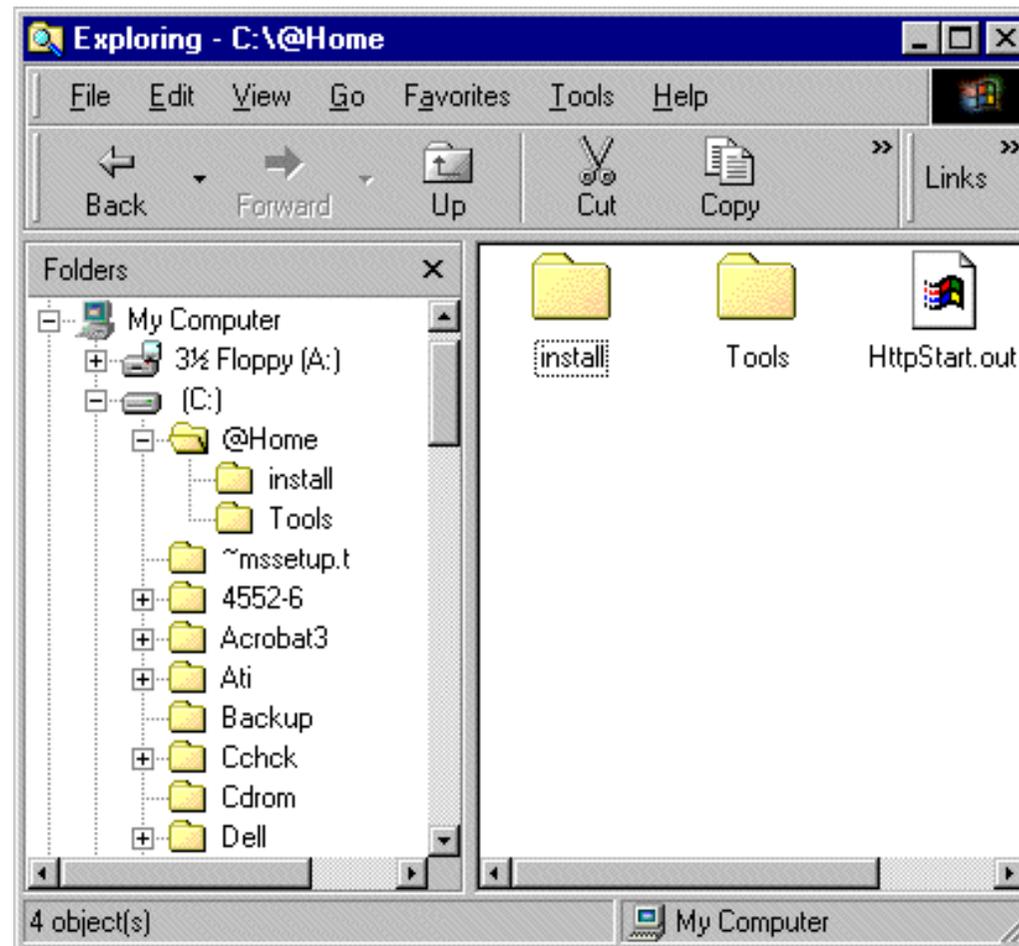
- **Windows 95** : *Août 1995*, cette version transfère des fonctionnalités de MS-DOS dans Windows





Historique – Evolution du Windows

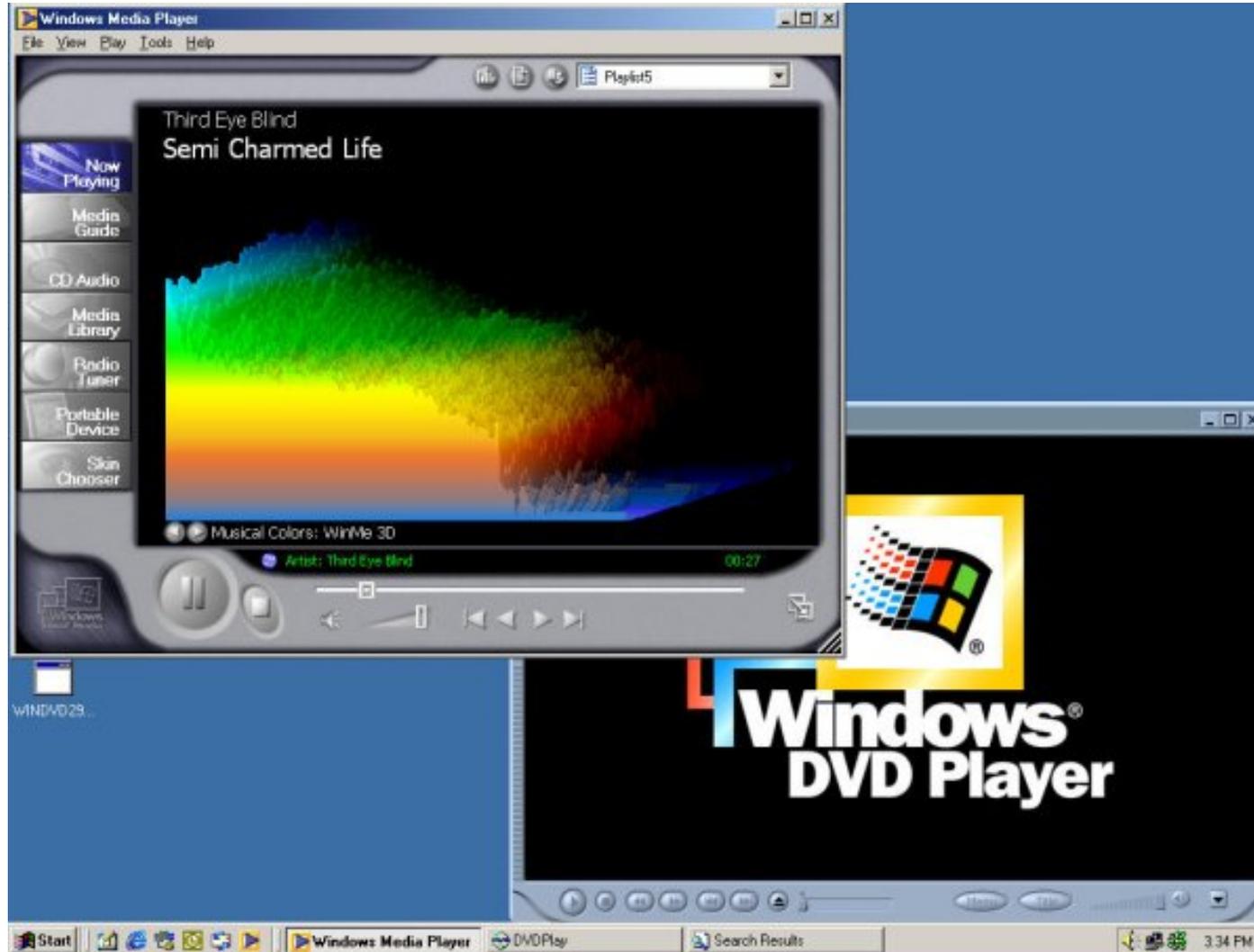
- **Windows 98** : *Juin 1998*, cette version intègre nativement d'autres fonctionnalités de DOS mais s'appuie toujours sur ce dernier





Historique – Evolution du Windows

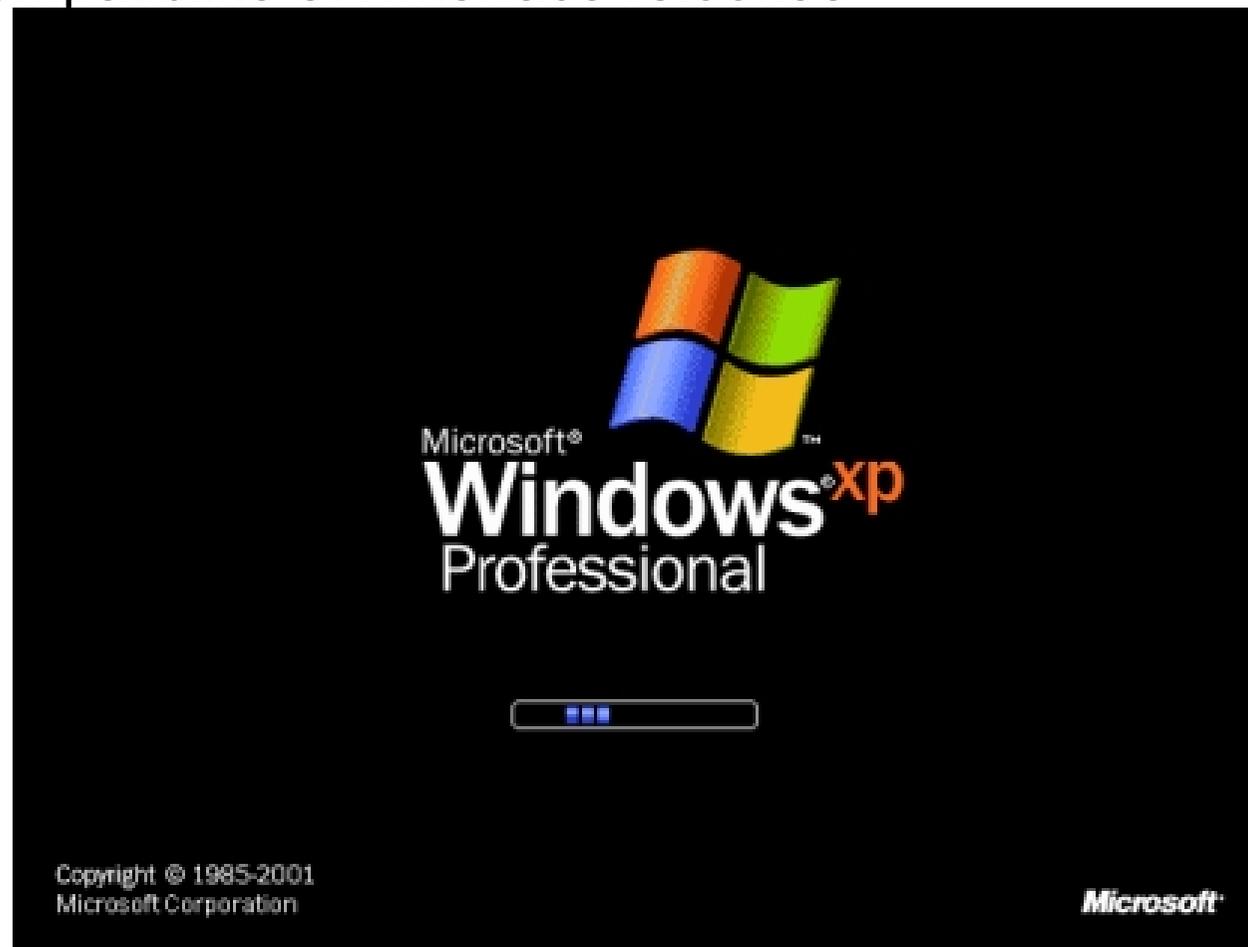
- **Windows ME** : *Septembre 2000*, Millennium Edition





Historique – Evolution du Windows

- **Windows XP** : *Octobre 2001*, Convergence Win 9x et Windows NT : la fiabilité de NT, la convivialité de Windows 9x, la compatibilité et l'interface relookée





Historique – Evolution du Windows

- **Windows Vista** : *Novembre 2006* pour les entreprises et *Janvier 2007* pour les particuliers





Historique – Evolution du Windows

- **Windows 7** *Sortie le 22 Octobre 2009*





Éléments de Windows XP

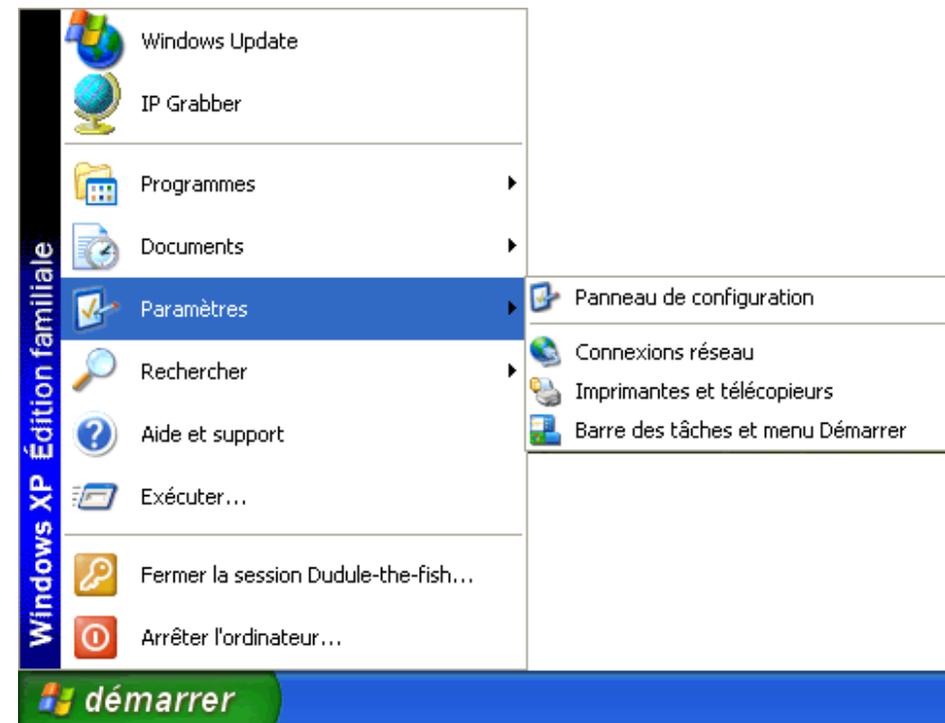
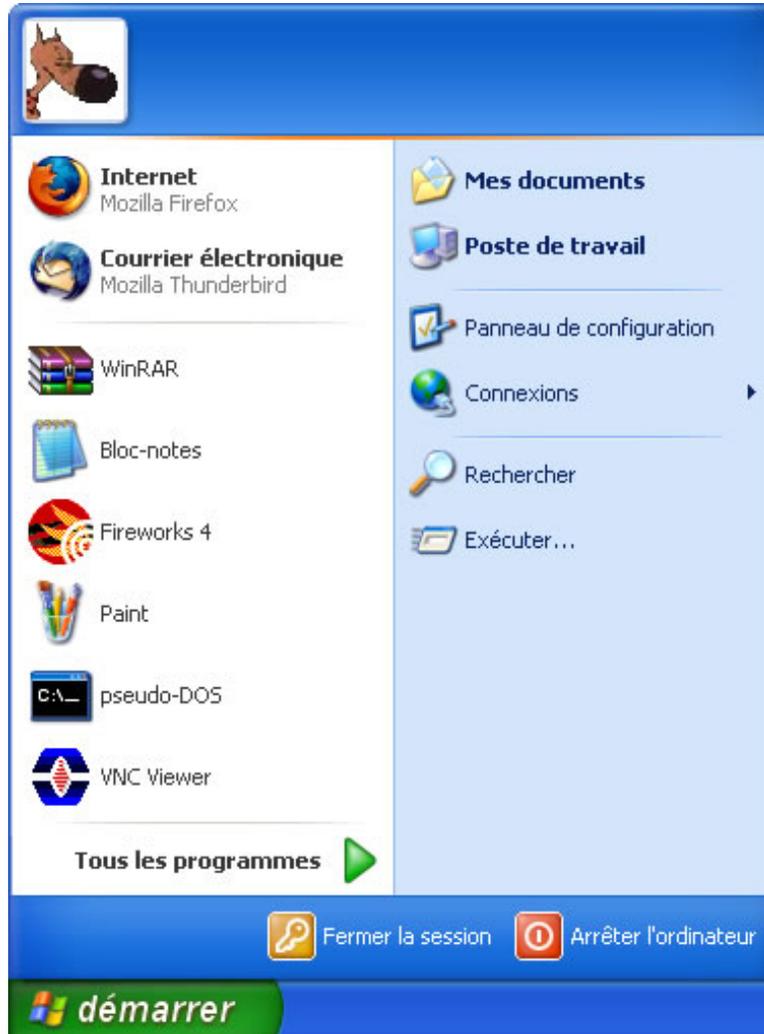
■ Le Bureau





Éléments de Windows XP

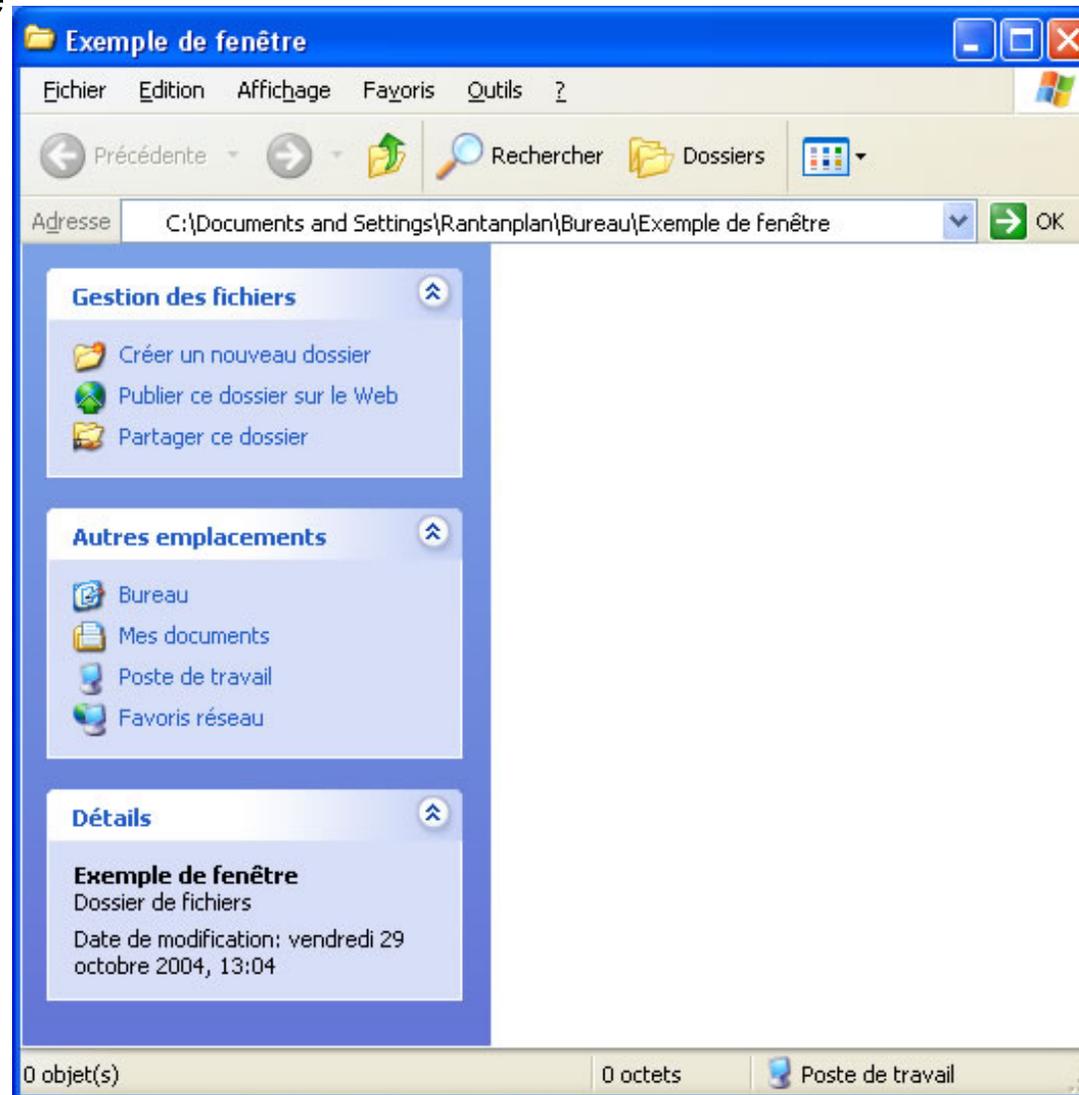
Le Menu Démarrer





Éléments de Windows XP

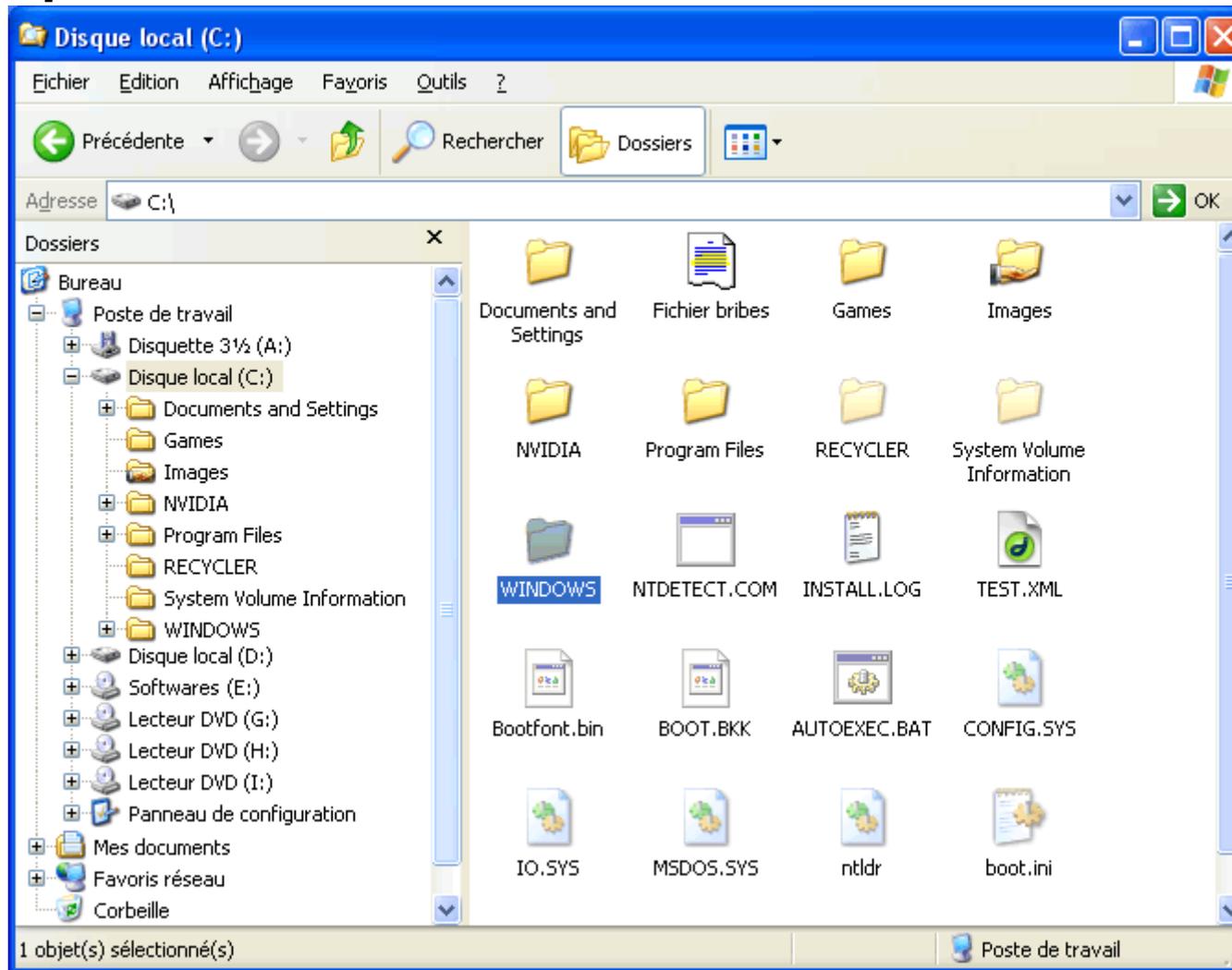
■ La Fenêtre





Éléments de Windows XP

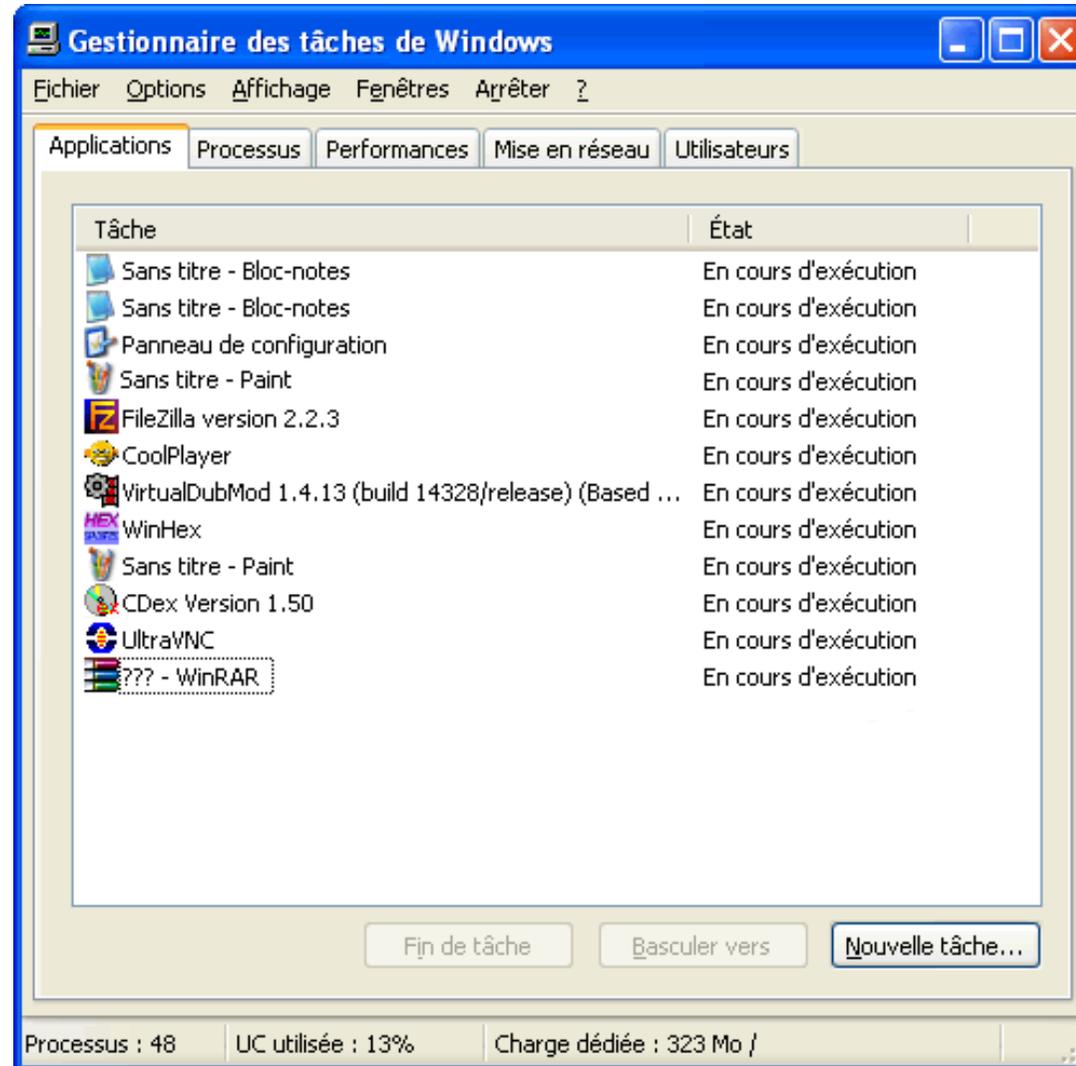
■ L'Explorateur Windows





Éléments de Windows XP

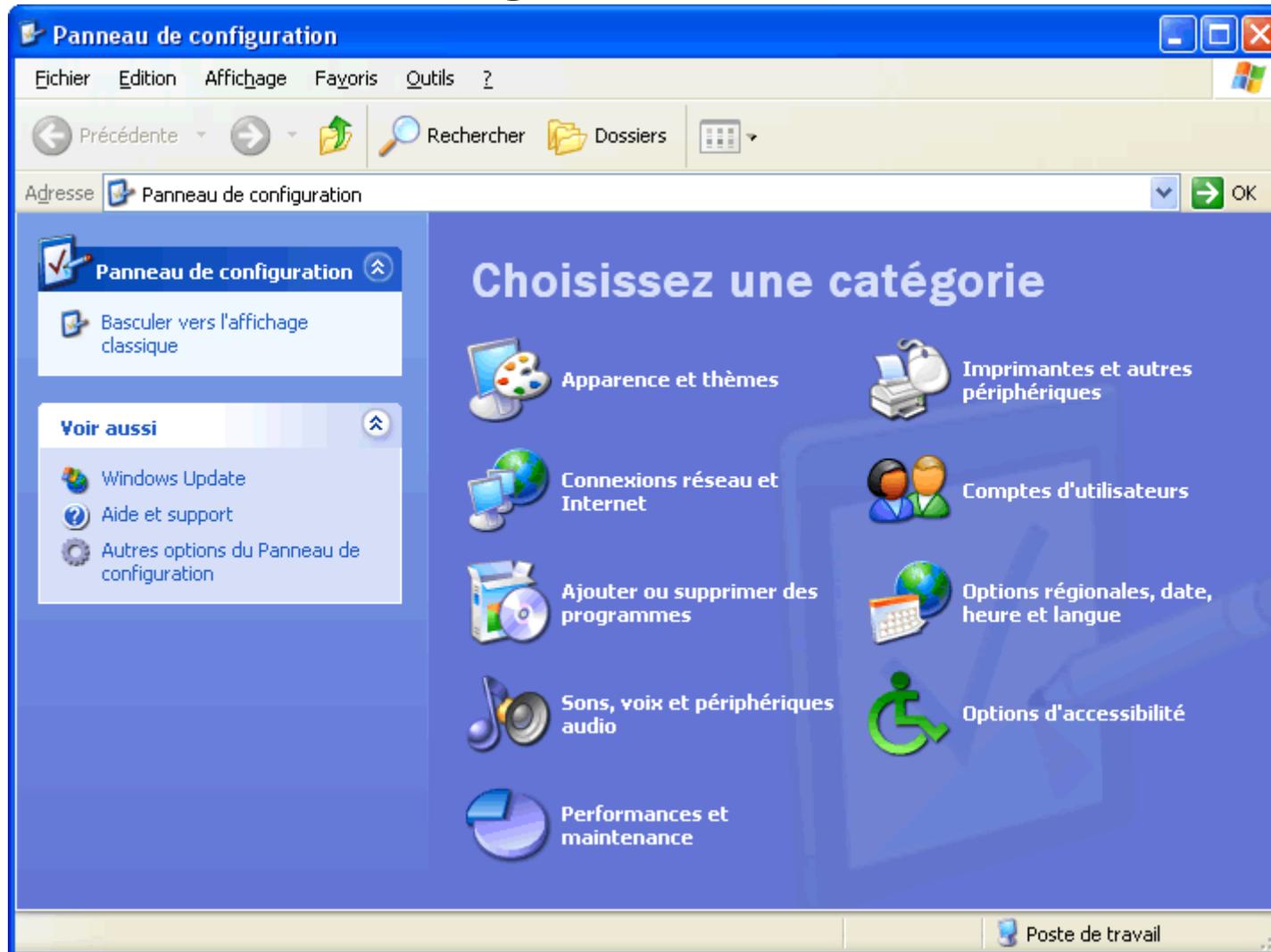
■ Le Gestionnaire de Tâches





Éléments de Windows XP

■ Le Panneau de Configuration





Cours N°3

Structures de Contrôles



1. Introduction

Définition :

Une structure de contrôle sert à contrôler le déroulement d'un traitement.

■ Un traitement peut s'exécuter de différentes manières:

✚ Séquentielle تسلسليا

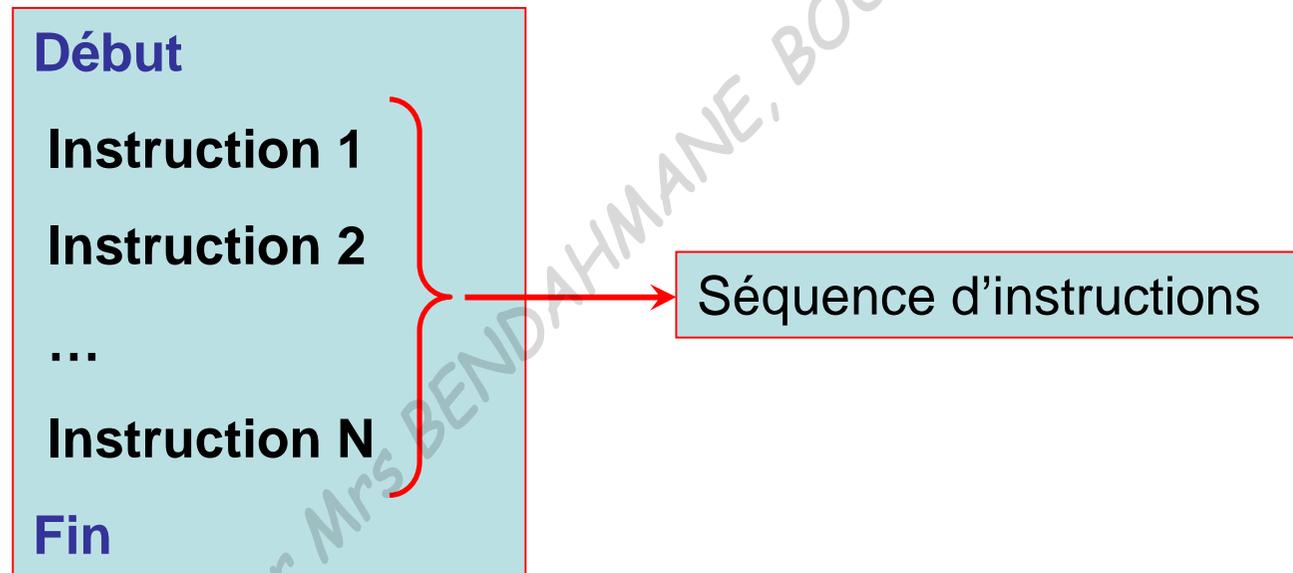
✚ Alternative (condition) تناوبيا

✚ Répétitive (boucle) تكريريا



2. Le traitement séquentiel

Le traitement séquentiel est une suite d'instruction qui s'exécutent l'une à la suite de l'autre.



Exemple: Addition de deux nombres A et B



2. Le traitement séquentiel (suite)

```
Program Addition;  
Uses wincrt;  
Var A,B,Somme:Real;  
  
Begin  
    Read(A);  
    Read(B);  
    Somme:=A+B;  
    Write(Somme);  
End.
```

Suite d'instructions
s'exécutant
séquentiellement



2. Le traitement séquentiel (suite)

Dans un programme, les instructions sont exécutées dans l'ordre de leur apparition, donc de façon séquentielle. Mais, **“l'intelligence”** d'un programme informatique provient essentiellement:

- De la possibilité de faire des choix entre plusieurs possibilités de traitement en fonction de différents critères (condition);
- De la possibilité d'exécuter rapidement une série d'instructions de façon répétitive (boucle).



3. Le traitement alternatif (structures de choix)

- L'instruction de choix permet la sélection entre deux possibilités (appelée sélection binaire).
- La condition s'exprime sous la forme:
 - d'une expression logique booléenne simple (condition simple).
 - ou combinée, plusieurs conditions composées avec des opérations logiques ET, OU et NON.



3. Le traitement alternatif (suite)

A. Structure conditionnelle simple (if ... then)

La structure conditionnelle simple se présente sous la forme suivante:

SI (condition) alors
séquence
FSI {lire Fin SI}

Principe :

Si la condition est vérifiée (vraie) alors la séquence d'instructions s'exécute. Dans le cas contraire, ne rien faire.



3. Le traitement alternatif (suite)

A. Structure conditionnelle simple (if ... then)

Traduction en Pascal:

- Instruction simple:

```
IF <condition> then <séquence>;
```

- Bloc d'instructions:

```
IF <condition> then  
    begin  
        <séquence>  
    end;
```



3. Le traitement alternatif (suite)

A. Structure conditionnelle simple (if ... then)

Exemple: Comparaison de deux variables

Algorithme comparaison

Variables utilisées:

A, B : nombres réels

1) Début

2) Lire (A)

3) Lire (B)

4) SI $A > B$ ALORS Ecrire ('A est supérieure à B) FSI

5) Fin



3. Le traitement alternatif (suite)

Programme: Première écriture

```
Program comparaison ;  
Uses wincrt ;  
Var A,B : REAL ;  
  
BEGIN  
  
    READLN(A) ;  
  
    READLN(B) ;  
  
    IF A > B THEN WRITELN (A,' est superieure a ',B) ;  
  
END.
```



3. Le traitement alternatif (suite)

Programme: Deuxième écriture

```
Program comparaison ;  
  
Uses wincrt ;  
  
Var A,B : REAL ;  
  
BEGIN  
  
    READLN(A) ;  
  
    READLN(B) ;  
  
    IF A > B THEN  
  
        begin  
  
            WRITELN (A, ' est superieure a ',B) ;  
  
        end ;  
  
END.
```



3. Le traitement alternatif (suite)

B. Structure conditionnelle composée (if ... then ... else)

La structure conditionnelle composée se présente sous la forme suivante:

SI (condition) ALORS

séquence 1

SINON

séquence 2

FSI

Principe :

Si la condition est vérifiée alors la séquence1 s'exécute.
Dans le cas contraire, c'est la séquence2 qui va s'exécuter.



3. Le traitement alternatif (suite)

B. Structure conditionnelle composée (if ... then ... else)

Traduction en Pascal:

```
■ IF <condition> THEN <séquence1>
```

```
  ELSE <séquence2>;
```

```
■ IF <condition> THEN
```

```
  begin
```

```
    <séquence1>
```

```
  end
```

```
ELSE
```

```
  begin
```

```
    <séquence2>
```

```
end;
```



3. Le traitement alternatif (suite)

B. Structure conditionnelle composée (if ... then ... else)

Attention ! On ne pose **JAMAIS** un point virgule (;) avant un **ELSE**.



3. Le traitement alternatif (suite)

B. Structure conditionnelle composée (if ... then ... else)

Exemple: Comparaison de deux variables (amélioré)

Algorithme comparaison

Variables utilisées:

A, B : nombres réels

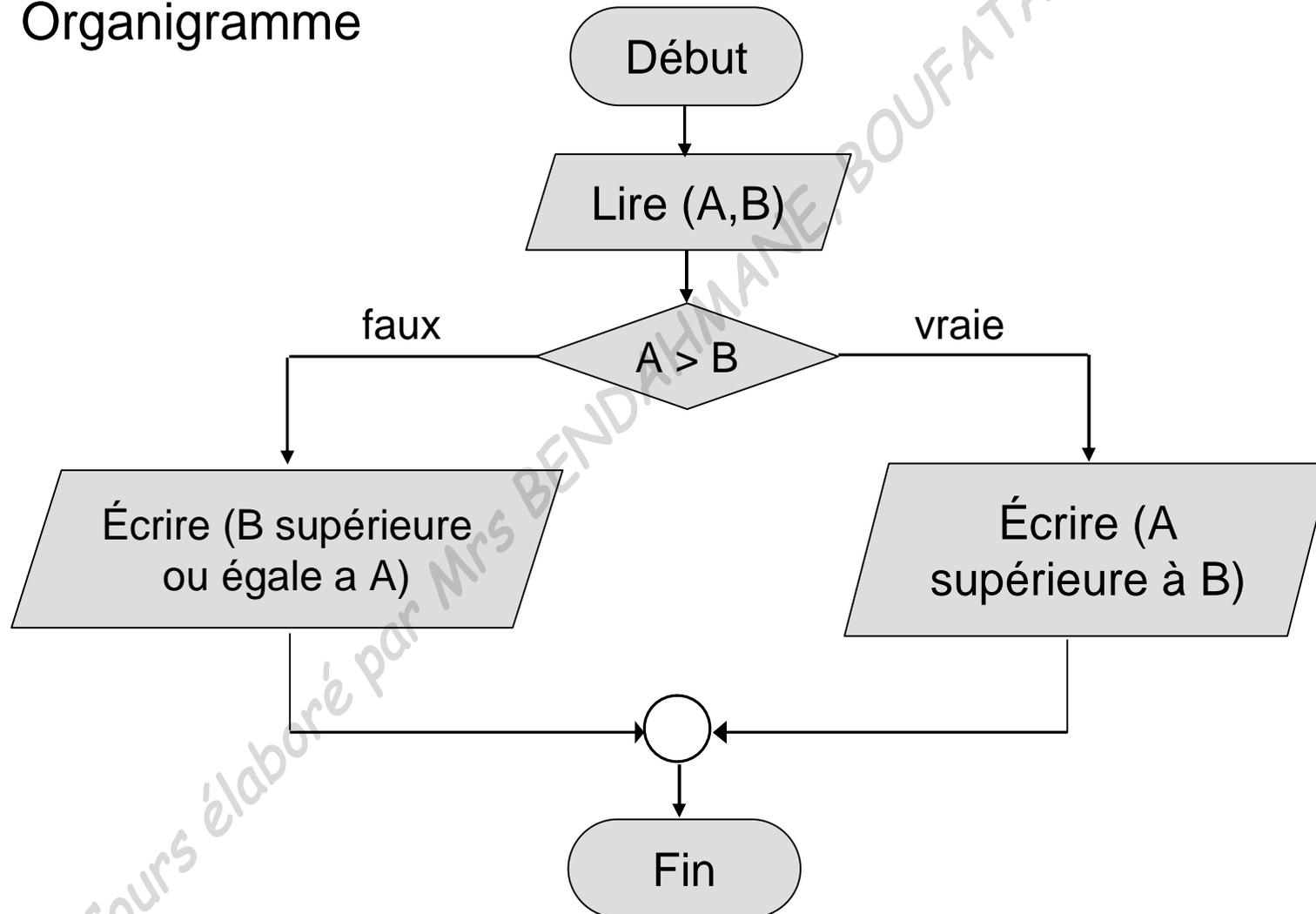
- 1) Début
- 2) Lire (A)
- 3) Lire (B)
- 4) SI $A > B$ ALORS Écrire ('A est supérieure à B')
- 5) SINON Écrire ('B est supérieure ou égale à A') FSI
- 6) Fin



3. Le traitement alternatif (suite)

B. Structure conditionnelle composée (if ... then ... else)

Organigramme





3. Le traitement alternatif (suite)

Programme: Première écriture

```
Program comparaison ;  
Uses wincrt ;  
Var A,B : REAL ;  
  
BEGIN  
  
    READLN(A) ;  
  
    READLN(B) ;  
  
    IF A > B THEN WRITELN (A,' est supérieure a ',B)  
    ELSE WRITELN (B,' est supérieure ou égale a ',A) ;  
  
END.
```



Programme: Deuxième écriture

```
Program comparaison ;
Uses wincrt;
Var A,B : REAL;
  BEGIN
    READLN(A);
    READLN(B);
    IF A > B THEN
      begin
        WRITELN (A,' est supérieure a ',B);
      end
    else
      begin
        WRITELN (B,' est supérieure a ',A);
      end;
    END.
```



3. Le traitement alternatif (suite)

C. Structure conditionnelle composée (case ... of)

Définition :

L'instruction CASE utilise la valeur d'une expression de type ordinal pour opérer le choix d'une instruction parmi d'autres (choix multiple) en vue de son exécution.

- Cette instruction compare la valeur d'une variable de type entier ou caractère à toute une série de valeur.
- Suivant la valeur effective de la variable différentes séquences d'instructions sont exécutées.
- Une séquence par défaut peut être prévue dans le cas où la variable n'est égale à aucune des valeurs énumérées.



3. Le traitement alternatif (suite)

C. Structure conditionnelle composée (case ... of)

AU CAS OU (**variable ou expression**)

choix 1 : <séquence 1>;

choix 2 : <séquence 2>;

...

choix n : <séquence n>;

SINON <séquence par défaut>;

FCAS

Remarque :

Cette structure de contrôle présente plusieurs cas d'exécutions du traitement mais un seul cas sera exécuté.



3. Le traitement alternatif (suite)

C. Structure conditionnelle composée (case ... of)

Traduction en Pascal:

■ Première écriture:

CASE <expression> **OF**

Choix 1 : <séquence 1>;

Choix 2 : <séquence 2>;

...

Choix n : <séquence n>;

END;



3. Le traitement alternatif (suite)

C. Structure conditionnelle composée (case ... of)

Traduction en Pascal:

■ Deuxième écriture:

CASE <expression> **OF**

Choix 1 : <séquence 1>;

Choix 2 : <séquence 2>;

...

Choix n : <séquence n>;

ELSE <séquence par défaut>;

END;



3. Le traitement alternatif (suite)

C. Structure conditionnelle composée (case ... of)

Exemple: Calcul du produit, somme et la moyenne de trois réelles suivant un choix

Algorithme menu

Variables utilisées:

NB1, NB2, NB3 : nombres réels

Choix : caractère

1) Début

2) Lire (NB1, NB2, NB3)

3) Lire (Choix)

4) AU CAS OU (Choix)

‘1’ : écrire (‘Le produit est : ’, $nb1 * nb2 * nb3$);

‘2’ : écrire (‘La somme est : ’, $nb1 + nb2 + nb3$);

‘3’ : écrire (‘la moyenne est : ’, $(nb1 + nb2 + nb3) / 3$);

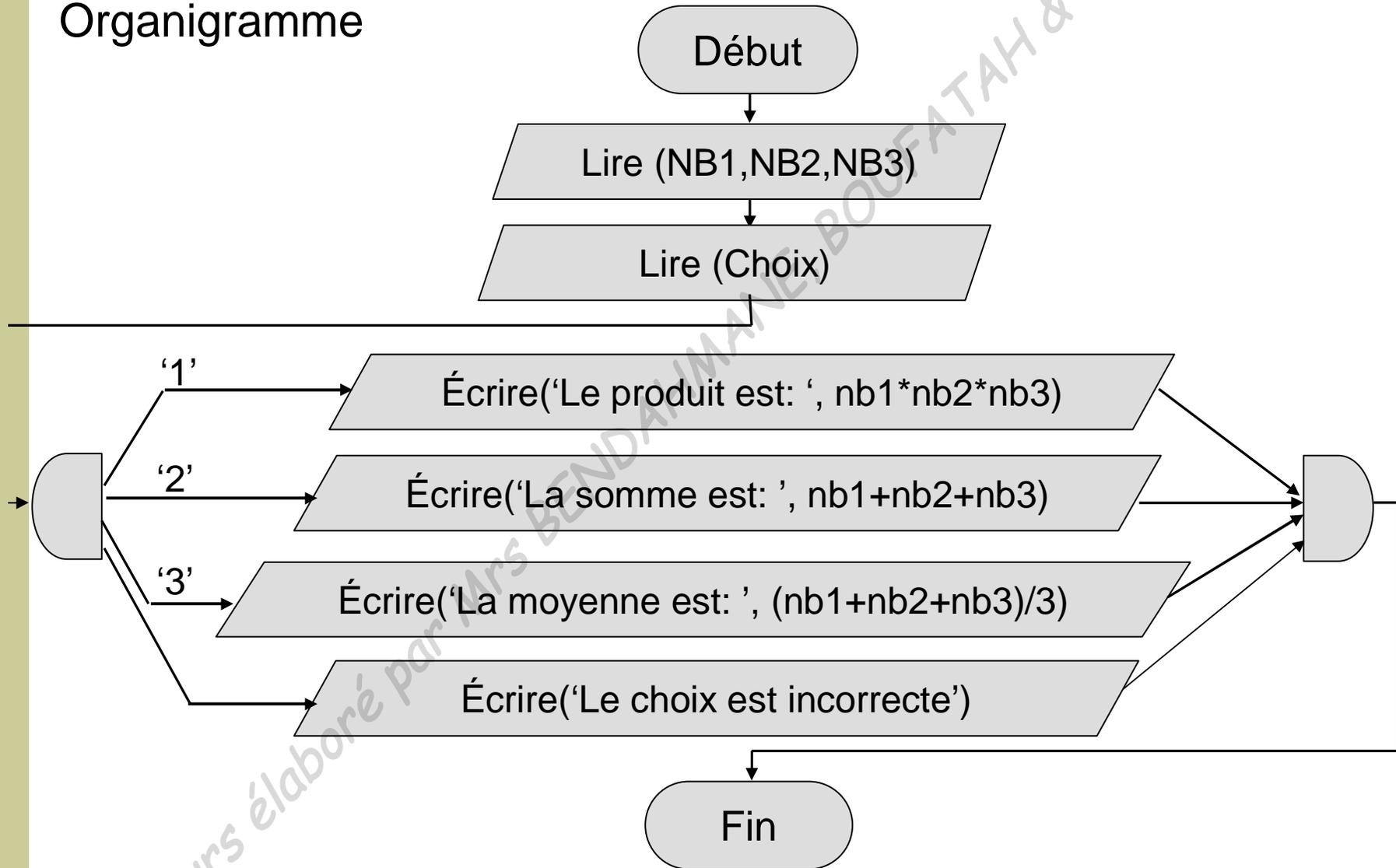
5) SINON écrire (‘Choix incorrecte) FCAS

6) Fin



3. Le traitement alternatif (suite)

Organigramme





3. Le traitement alternatif (suite)

Programme

```
Program menu ;
Uses wincrt;
Var NB1,NB2,NB3 : REAL;
    Choix : CHAR;
BEGIN
    WRITELN('Entrer trois nombres :');
    READLN(NB1,NB2,NB3);
    WRITELN('1 pour la multiplication. ');
    WRITELN('2 pour la somme. ');
    WRITELN('3 pour la moyenne. ');
    READLN(Choix);
    CASE Choix of
        '1': WRITELN ('Le produit est : ',NB1*NB2*NB3);
        '2': WRITELN ('La somme est : ',NB1+NB2+NB3);
        '3': WRITELN ('La moyenne est : ',(NB1+NB2+NB3)/3);
        ELSE WRITELN ('Le choix est incorrecte');
    END;
END;
END.
```



3. Le traitement alternatif (suite)

C. Structure conditionnelle composée (case ... of)

Remarque:

- Les valeurs d'une instruction CASE sont des constantes mais elles peuvent définir un domaine (intervalle). Exemple:

CASE L OF

5 .. 20 : <instruction 1 >;

30 .. 40 : <instruction 2>;

END;



4. Le traitement répétitif (les boucles)

Dans un programme, il arrive souvent qu'une ou plusieurs instructions doivent être exécuter plusieurs fois dans une structure répétitive appelée boucle.

Dans une boucle, le nombre de répétitions:

- Peut être connu, fixé à l'avance.
- Il peut dépendre d'une condition permettant l'arrêt et la sortie de cette boucle.

On dispose de trois structures pour contrôler un traitement répétitif:

- La boucle "*Tant que ... faire*".
- La boucle "*Pour ... faire*".
- La boucle "*Répéter ... faire*".



4. Le traitement répétitif (suite)

A. La boucle "Tant que ... faire" (while ... do)

La boucle "*Tant que ... faire*" se présente sous la forme suivante:

Tant que (condition) faire
séquence
Fin tant que

Principe :

Tant que la condition est vérifiée la séquence s'exécute. Elle ne s'arrêtera que lorsque la condition est invérifiable.



4. Le traitement répétitif (suite)

A. La boucle "Tant que ... faire" (while ... do)

Traduction en Pascal:

■ Première écriture:

```
WHILE <condition> DO <séquence>;
```

■ Deuxième écriture

```
WHILE <condition> DO
```

```
begin
```

```
    <séquence>;
```

```
end;
```



4. Le traitement répétitif (suite)

A. La boucle "Tant que ... faire" (while ... do)

Exemple: Afficher les 100 premiers nombre entiers positifs

Algorithme premier_N_E

Variables utilisées:

I : nombre entier

1) Début

2) I=0

3) Tant que I <= 100 faire

4) Écrire(I)

5) I=I+1 Fin tant que

6) Fin



4. Le traitement répétitif (suite)

A. La boucle "Tant que ... faire" (while ... do)

Programme

```
Program premier_N_E;  
Uses wincrt;  
Var   I:Integer;  
  
BEGIN  
    I:=0; {Initialisation}  
    WHILE I<=100 DO  
        begin  
            WRITELN(I);  
            I:=I+1; {Incrémentation}  
        end;  
    END.
```



4. Le traitement répétitif (suite)

A. La boucle "Tant que ... faire" (while ... do)

Attention ! Il faut bien choisir le test d'arrêt de la boucle sinon on obtient une *boucle infinie*.

Exemple: calcul du carré d'un nombre réel.

```
r := 1.5;           {initialisation du nombre réel}
  WHILE r > 0 DO    {condition toujours vraie ! étant donné l'initialisation}
    r := SQR(r);    {calcul du carré avec la fonction prédéfini SQR}
```

C'est une **boucle infinie** parce que le carré d'un nombre renvoie toujours une valeur positive. La condition $r > 0$ est toujours vraie.



4. Le traitement répétitif (suite)

B. La boucle "Pour ... faire" (For ... To ... do)

La boucle "*Pour ... faire*" se présente sous la forme suivante:

```
Pour compteur = <valeur initiale> à <valeur finale> faire  
  
    <séquence>  
  
Fin Pour
```

Principe :

Pour la variable de contrôle allant de la valeur initiale jusqu'à la valeur finale, exécuter la séquence d'instruction.



4. Le traitement répétitif (suite)

B. La boucle "Pour ... faire" (For ... To ... do)

Traduction en Pascal:

■ Première écriture:

```
FOR <compteur>:= <valeur initiale> TO <valeur finale> DO  
    <séquence>;
```

■ Deuxième écriture

```
FOR <compteur>:= <valeur initiale> TO <valeur finale> DO  
    begin  
        <séquence>;  
    end;
```



4. Le traitement répétitif (suite)

B. La boucle "Pour ... faire" (For ... To ... do)

Exemple: Afficher les 100 premiers nombre entiers positifs

Algorithme premier_N_E

Variables utilisées:

I : nombre entier

1) Début

2) Pour I = 0 à 100 faire

3) Écrire(I)

4) Fin Pour

5) Fin



4. Le traitement répétitif (suite)

B. La boucle "Pour ... faire" (For ... To ... do)

Programme

```
Program premier_N_E;  
Uses winCRT;  
Var I : Integer;  
  
BEGIN  
    FOR I:=0 TO 100 DO  
        begin  
            WRITELN(I);  
        end;  
END.
```



4. Le traitement répétitif (suite)

C. La boucle "Répéter ... jusqu'à" (Repeat ... until)

La boucle "*Répéter ... jusqu'à*" se présente sous la forme suivante:

Répéter

<séquence>

Jusqu'à <condition vérifiée>

Principe :

La séquence d'instructions délimitée par les mots **Répéter** et **Jusqu'à** est exécutée jusqu'à ce que la condition évaluée à la fin de la boucle soit vraie.



4. Le traitement répétitif (suite)

C. La boucle "Répéter ... jusqu'à" (Repeat ... until)

Traduction en Pascal:

REPEAT

<séquence>;

UNTIL <condition>;

Cours élaboré par Mrs BENDAHMANE, BOUFATAH & BRAHMI



4. Le traitement répétitif (suite)

C. La boucle "Répéter ... jusqu'à" (Repeat ... until)

Exemple: Afficher les 100 premiers nombre entiers positifs

Algorithme premier_N_E

Variables utilisées:

I : nombre entier

1) Début

2) $I = 0$

3) Répéter

4) Écrire(I)

5) $I = I + 1$

6) Jusqu'à $I > 100$

7) Fin



4. Le traitement répétitif (suite)

C. La boucle "Répéter ... jusqu'à" (Repeat ... until)

Programme

```
Program premier_N_E ;  
Uses wincrt ;  
Var I : Integer ;  
  
BEGIN  
    I:=0 ;  
    repeat  
        WRITELN(I) ;  
        I:=I+1 ;  
    until I > 100 ;  
END.
```



4. Le traitement répétitif (suite)

D. Les boucles imbriquées

Les boucles peuvent être imbriquées les unes dans les autres. Une boucle "*tant que ... faire*" peut contenir une autre boucle "*tant que ... faire*", ou une autre boucle "*répéter ... jusqu'à*", ou une autre boucle "*pour ... faire*".

Autrement dit, une boucle peut contenir une autre boucle qui elle-même peut contenir une autre boucle ainsi de suite.



5. Remarques générales sur les boucles

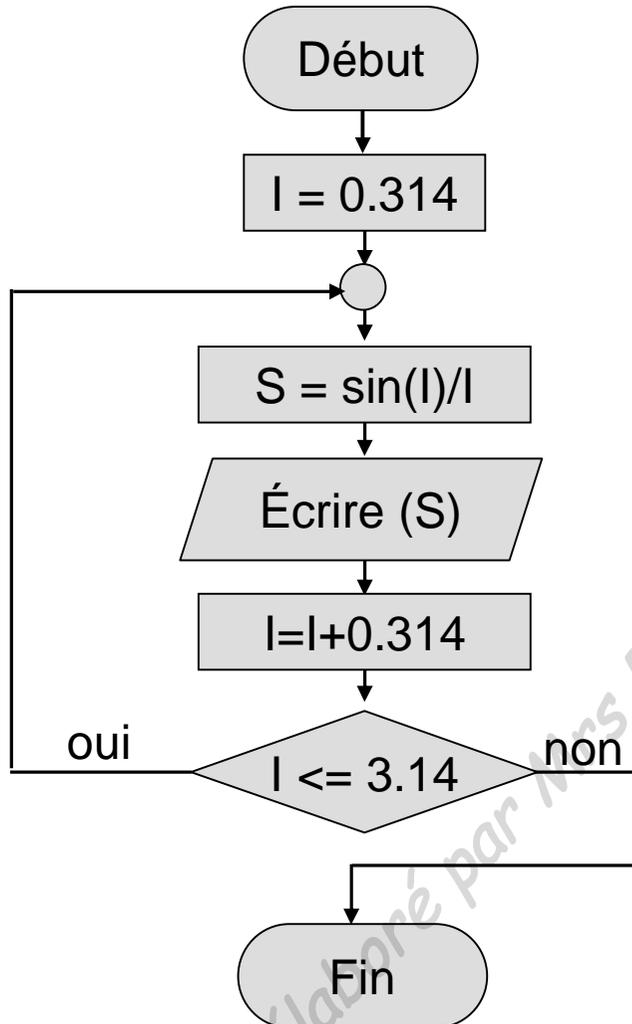
- 1) La boucle "*while*" peut ne pas s'exécuter du tout si la condition d'accès à la boucle n'est pas vérifiée.
- 2) La boucle "*repeat*" s'exécute au moins une fois.
- 3) La boucle "*for*" s'exécute toujours, le nombre d'exécution de la séquence (itérations) est connu d'avance.
- 4) Les boucles "*while*" et "*repeat*" nécessitent toujours une initialisation et une incrémentation du compteur de la boucle.
- 5) La boucle "*for*" est utilisée exclusivement dans le cas où le nombre d'itérations est connu.
- 6) Si le nombre d'instructions de la boucle est supérieur à 1 celles-ci doivent être mises entre un **BEGIN** et un **END**.



6. L'instruction GOTO et la déclaration d'étiquette

En théorie, les structures de choix et de répétition proposées par le Pascal sont suffisantes pour traduire toutes les situations possibles. Mais, le Pascal dispose également d'une instruction de branchement inconditionnel GOTO.

Exemple:



```
Program sinus;  
Uses wincrt;  
Var i,s : real;  
Label 10;  
BEGIN  
    I:=0.314;  
10 : S:=sin(I)/I;  
    WRITELN(S);  
    I:=I+0.314;  
    IF I <= 3.14 then GOTO 10;  
END.
```



Cours N°4

Les Matrices

Cours élaboré par Mrs BENDAHMANE, BOUFATAH & BRAHMI



1. Introduction

Définition :

Une matrice $n \times m$ est un tableau de nombres à n lignes et m colonnes. n et m sont les dimensions de la matrice.

Exemple avec $n = 2$ et $m=3$

$$A = \begin{bmatrix} 1 & -5 & 3 \\ 2 & 1 & 4 \end{bmatrix}$$



1. Introduction (suite)

- Dans une matrice \mathbf{A} , on note \mathbf{A}_{ij} l'élément situé à l'intersection de la ligne i et de la colonne j (la ligne est toujours nommée en premier).

$$\mathbf{A} = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1m} \\ A_{21} & A_{22} & \cdots & A_{2m} \\ \cdots & \cdots & \cdots & \cdots \\ A_{n1} & A_{n2} & \cdots & A_{nm} \end{bmatrix}$$

- On note $[\mathbf{A}_{ij}]$ la matrice d'élément général \mathbf{A}_{ij} . On a donc: $\mathbf{A} = [\mathbf{A}_{ij}]$



1. Introduction (suite)

- Si $m = 1$ ou $n = 1$, la matrice est appelée **vecteur**. Plus précisément:
 - Si $m = 1$, la matrice est un **vecteur-colonne**.
 - Si $n = 1$, la matrice est un **vecteur-ligne**.
- Si $m = n$, la matrice est appelée **matrice carré**.

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{bmatrix}$$

$$x = [x_1 \quad x_2 \quad \dots \quad x_n]$$



2. Quelques matrices carrées particulières

A. Matrice diagonale

$$D = \begin{bmatrix} D_{11} & 0 & 0 & 0 \\ 0 & D_{22} & 0 & 0 \\ 0 & 0 & D_{33} & 0 \\ 0 & 0 & 0 & D_{44} \end{bmatrix}$$



2. Quelques matrices carrées particulières(suite)

B. Matrice unité

$$I = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



2. Quelques matrices carrées particulières(suite)

C. Matrice triangulaire supérieure

$$U = \begin{bmatrix} U_{11} & U_{12} & U_{13} & U_{14} \\ 0 & U_{22} & U_{23} & U_{24} \\ 0 & 0 & U_{33} & U_{34} \\ 0 & 0 & 0 & U_{44} \end{bmatrix}$$



2. Quelques matrices carrées particulières(suite)

D. Matrice triangulaire inférieure

$$L = \begin{bmatrix} L_{11} & 0 & 0 & 0 \\ L_{21} & L_{22} & 0 & 0 \\ L_{31} & L_{32} & L_{33} & 0 \\ L_{41} & L_{42} & L_{43} & L_{44} \end{bmatrix}$$



2. Quelques matrices carrées particulières(suite)

E. Matrice Symétrique

Une matrice carrée A est dite symétrique si: $A_{ji} = A_{ij}$.

$$A = \begin{bmatrix} 6 & 1 & 2 & 3 \\ 1 & 5 & -2 & 3 \\ 2 & -2 & 5 & -4 \\ 3 & 3 & -4 & 6 \end{bmatrix}$$



3. Opérations sur les matrices

A. Addition, soustraction

L'addition et la soustraction des matrices se font terme à terme. Les matrices doivent avoir les même dimensions:

$$\begin{bmatrix} 4 & 6 & 2 \\ 0 & 1 & 3 \end{bmatrix} + \begin{bmatrix} -1 & 3 & 6 \\ 2 & -5 & 12 \end{bmatrix} = \begin{bmatrix} 3 & 9 & 8 \\ 2 & -4 & 15 \end{bmatrix}$$

$$\begin{bmatrix} 4 & 6 & 2 \\ 0 & 1 & 3 \end{bmatrix} - \begin{bmatrix} -1 & 3 & 6 \\ 2 & -5 & 12 \end{bmatrix} = \begin{bmatrix} 5 & 3 & -4 \\ -2 & 6 & -9 \end{bmatrix}$$



3. Opérations sur les matrices (suite)

B. Multiplication par un nombre

Chaque terme de la matrice est multiplié par un nombre:

$$3 \times \begin{bmatrix} 4 & 6 & 2 \\ 0 & 1 & 3 \end{bmatrix} = \begin{bmatrix} 12 & 18 & 6 \\ 0 & 3 & 9 \end{bmatrix}$$



3. Opérations sur les matrices (suite)

C. Transposition

La transposée A^T d'une matrice A est la matrice obtenue en échangeant les lignes et les colonnes de A :

$$A = \begin{bmatrix} 4 & 6 & 2 \\ 0 & 1 & 3 \end{bmatrix} \Leftrightarrow A^T = \begin{bmatrix} 4 & 0 \\ 6 & 1 \\ 2 & 3 \end{bmatrix}$$

La transposée d'un vecteur-colonne est un vecteur-ligne.



3. Opérations sur les matrices (suite)

D. Multiplication des matrices

Le produit de la matrice $\mathbf{A}(n \times m)$ par la matrice $\mathbf{B}(m \times p)$ est la matrice $\mathbf{C}(n \times p)$ telle que l'élément C_{ij} est égal au produit scalaire de la ligne i de la matrice \mathbf{A} par la colonne j de la matrice \mathbf{B} :

$$C_{ij} = \sum_{k=1}^m A_{ik} \cdot B_{kj} \quad i = 1..n \quad j = 1..p$$

$$\begin{bmatrix} 4 & 6 & 2 \\ 0 & 1 & 3 \end{bmatrix} \bullet \begin{bmatrix} 2 & 0 \\ 3 & -1 \\ 4 & 3 \end{bmatrix} = \begin{bmatrix} 34 & 0 \\ 15 & 8 \end{bmatrix}$$



3. Opérations sur les matrices (suite)

E. Déterminant d'une matrice carrée 2×2

- Le déterminant *détermine* l'unicité de la solution d'un système d'équations linéaires.
- Le déterminant d'une matrice carrée 2×2 est la quantité:

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \Rightarrow \det(A) = |A| = \begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - bc$$

- La matrice A est singulière si $\det(A) = 0$, régulière dans le cas contraire.



3. Opérations sur les matrices (suite)

F. Déterminant d'une matrice carrée 3×3

Le déterminant peut se calculer de manière récursive en développant, par exemple, par rapport à la première ligne:

$$\begin{vmatrix} a & b & c \\ d & e & f \\ g & h & i \end{vmatrix} = a \begin{vmatrix} e & f \\ h & i \end{vmatrix} - b \begin{vmatrix} d & f \\ g & i \end{vmatrix} + c \begin{vmatrix} d & e \\ g & h \end{vmatrix}$$

$$= a(ei - fh) - b(di - fg) + c(dh - eg)$$

$$= aei - afh - bdi + bfg + cdh - ceg$$



Cours N°5

MS Office 2003: POWERPOINT





C'est quoi PowerPoint?

- **PowerPoint fait partie de la suite bureautique MS Office éditée par la société Microsoft.**
- **C'est un logiciel classé dans la catégorie des logiciels de Pré – PAO (*Présentation Assistée par Ordinateur*).**
- **Il permet de concevoir, mettre en forme, valoriser les idées qui seront développées lors d'une Présentation (réunion, soutenance, meeting ...).**



C'est quoi PowerPoint?

- **Ce logiciel est capable de gérer des documents aussi variés que :**
 - ❖ **Document A4,**
 - ❖ **Diapositive,**
 - ❖ **Transparents,**
 - ❖ **Rétro-projection directe d'un écran d'ordinateur.**

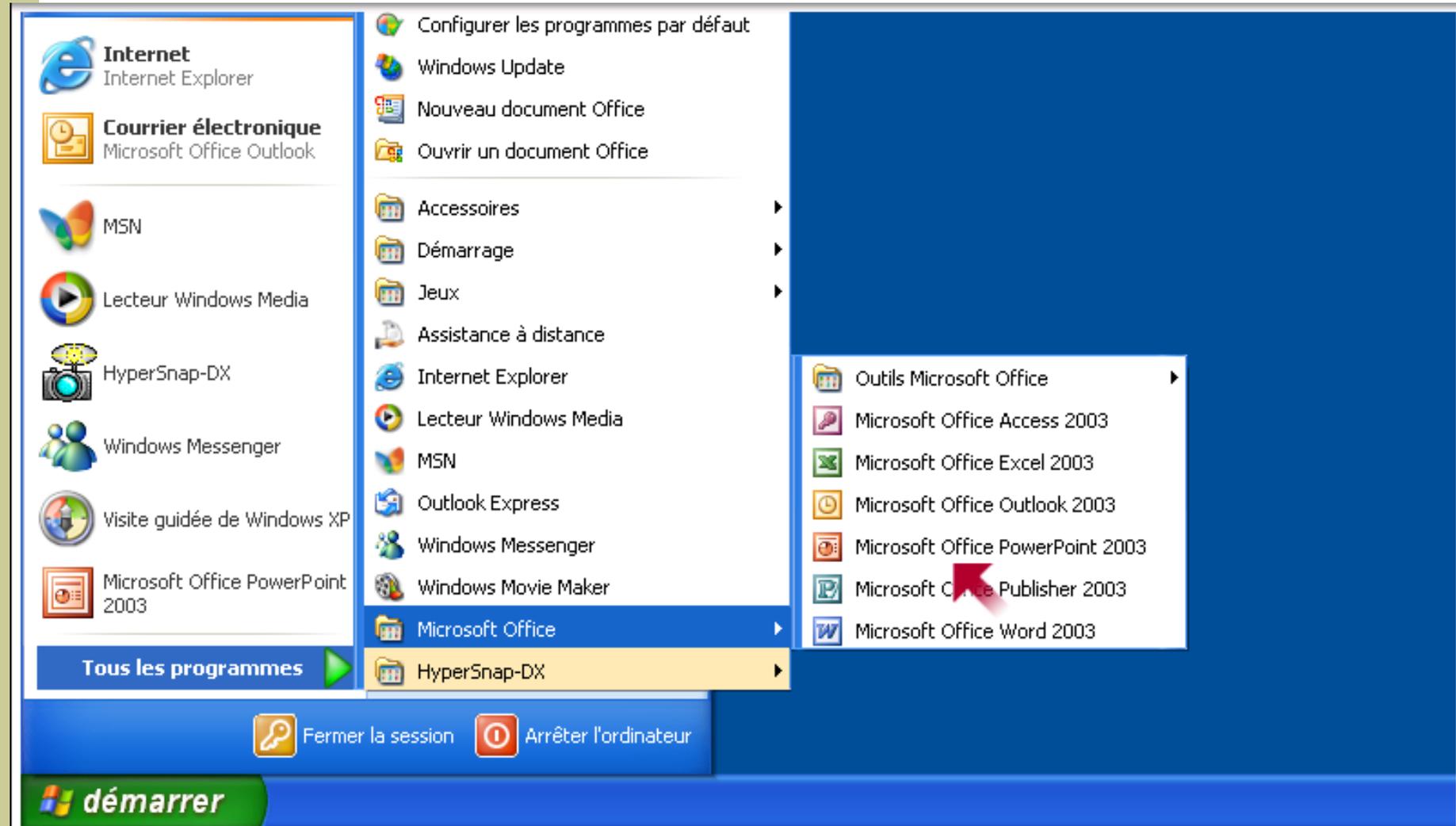


Ce que vous pouvez accomplir avec PowerPoint

- **Créer une présentation sur papier ou la présenter sur ordinateur**
- **Organiser vos idées pour la présentation**
- **Embellir votre présentation pour lui donner une allure "professionnelle"**
- **Ajouter des graphiques, tableaux, organigrammes et autres objets pour améliorer la compréhension des idées que vous voulez présenter**

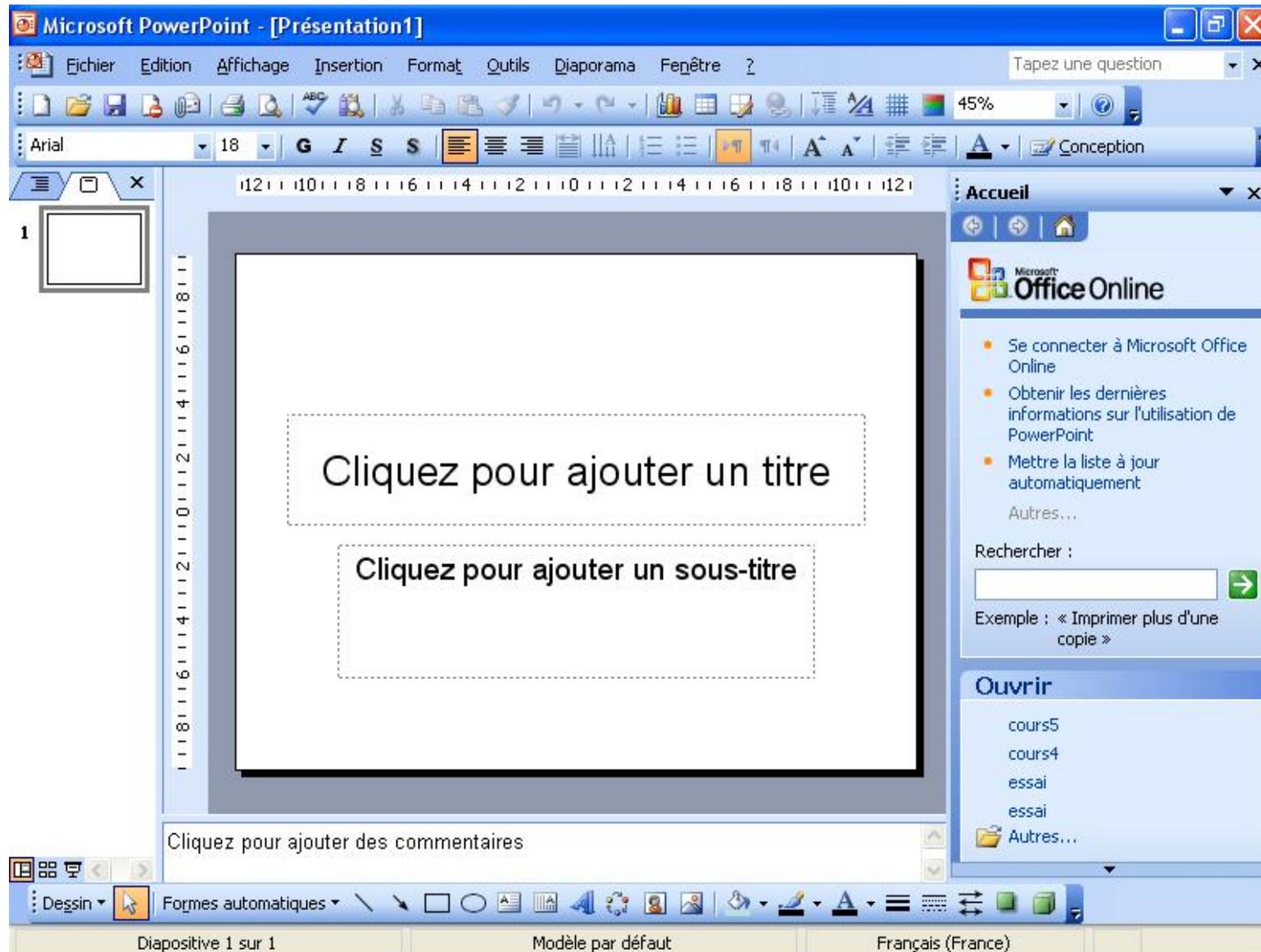


Lancement de l'application PowerPoint





Patientez quelques secondes pendant le chargement de PowerPoint et vous verrez apparaître une fenêtre de ce genre :





L'interface PowerPoint

The screenshot shows the Microsoft PowerPoint 2003 interface. The title bar reads "Microsoft PowerPoint - [Présentation1]". The menu bar includes "Fichier", "Edition", "Affichage", "Insertion", "Format", "Outils", "Diaporama", and "Fenêtre". The toolbar contains various icons for file operations and editing. The main workspace shows a slide with a title box containing "Cliquez pour ajouter un titre" and a subtitle box containing "Cliquez pour ajouter un sous-titre". A status bar at the bottom indicates "Diapositive 1 sur 1", "Modèle par défaut", and "Français (France)".

Annotations with red arrows point to the following elements:

- Volet Office**: Points to the Office Ribbon (Accueil) on the right side of the window.
- Volet diapositives en miniatures**: Points to the slide thumbnail pane on the left side of the window.
- La diapositive**: Points to the main slide area.
- Mode d'affichage (Mode plan - Mode trieuse - Mode diaporama)**: Points to the view selection icons at the bottom left of the window.



Modes d’Affichage PowerPoint

Pour vous aider à la création de votre présentation, PowerPoint sépare les fonctions du logiciel en 3 "modes" :

- *Mode Normal*
- *Mode Trieuse de Diapositives*
- *Mode Diaporama*

Vous pouvez passer d'un mode à l'autre en utilisant les boutons que vous retrouvez au coin inférieur gauche de l'écran de PowerPoint  ou en utilisant le menu *Affichage* et l'une des 3 premières options du menu



Modes d’Affichage PowerPoint



Mode Normal

- C'est une combinaison des modes Diapositive et Plan
- Il permet la création des pages ou des "diapositives" tout en voyant la structure de votre présentation
- Vous pouvez changer tous les aspects des diapositives (mise en page, les couleurs et la taille du texte, ajouter ou enlever des graphiques, des tableaux, ...)
- Vous pouvez aussi ajouter le texte structuré qui composera votre présentation
- Vous pouvez en tout temps changer l'ordre ou l'importance des idées



Modes d’Affichage PowerPoint



Mode Trieuse de Diapositives

- Ce mode permet d'ajouter des effets de transition (pour le passage d'une diapositive à une autre) à votre présentation si vous songez la présenter en utilisant un ordinateur
- Vous pouvez chronométrer le temps que vous prenez pour donner votre présentation. Vous pourrez ainsi ajuster votre présentation selon le temps qui vous est alloué



Modes d’Affichage PowerPoint



Mode Diaporama

- Ce mode affiche votre présentation à l'écran de l'ordinateur. Même dans ce mode, vous avez accès à des options pour mieux présenter



Définitions

Qu'est-ce qu'une présentation ?

Une **présentation** est une séquence d'images appelées *diapositives*.

Qu'est-ce qu'une diapositive ?

Une **diapositive** est assimilable à **une** page de document contenant du texte, des images, des graphiques, des dessins, du son, des vidéo. Le terme **diapositive** définit un visuel qui est indépendant de l'utilisation qui en sera faite : affichage à l'écran ou impression sur papier ou transparent.



Définitions

Qu'est-ce qu'un diaporama ?

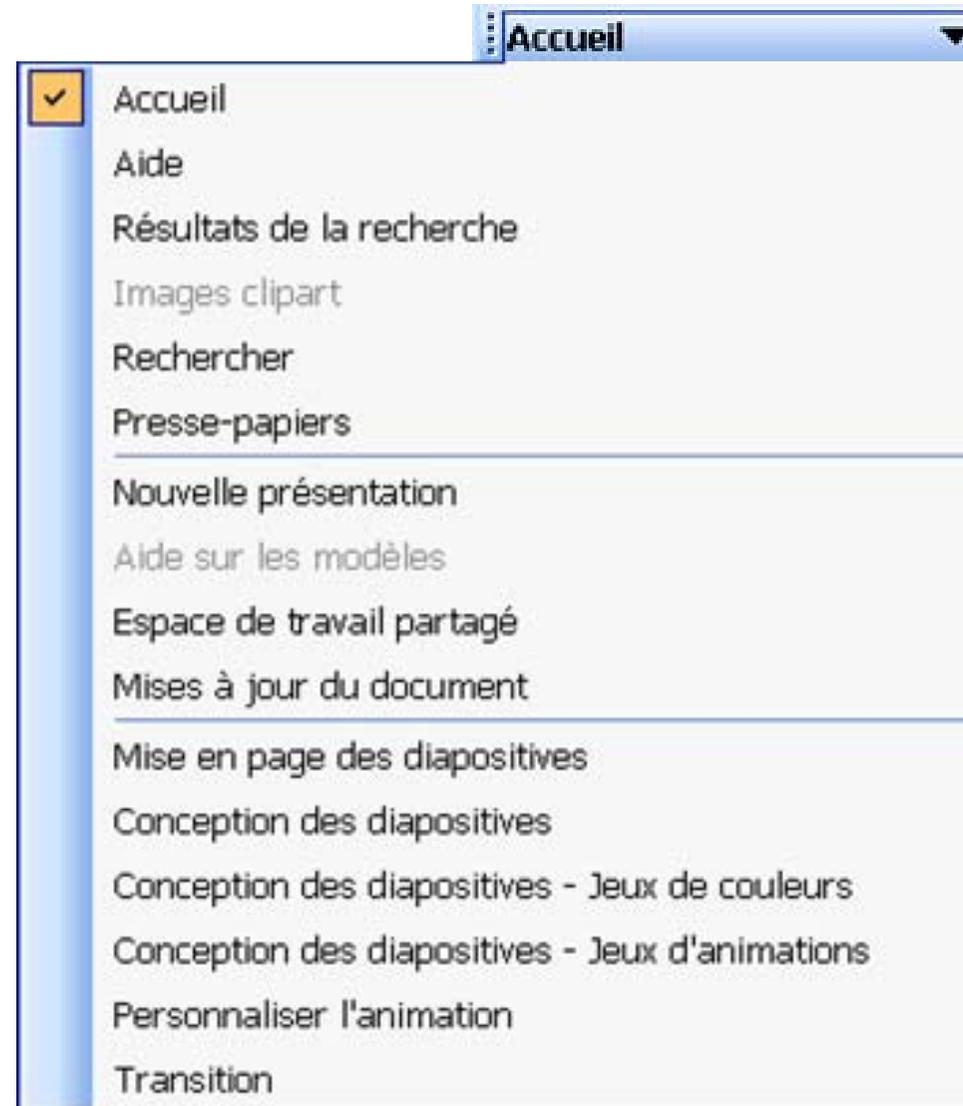
Un **diaporama** est un spectacle de projection de **diapositives** ; par extension on entend par ce terme toute suite d'images ou de documents reliés par des effets et, sur lesquels il est possible de mettre du son.

Toutes les diapositives d'une **présentation** Powerpoint sont contenues dans un seul et même fichier portant l'extension **.ppt**. Si la **présentation** est enregistrée sous forme de **diaporama**, l'extension du fichier sera **.pps**.



Volet Office

Il regroupe les options des assistants ainsi que les options, ou propriétés, des objets placés dans votre présentation



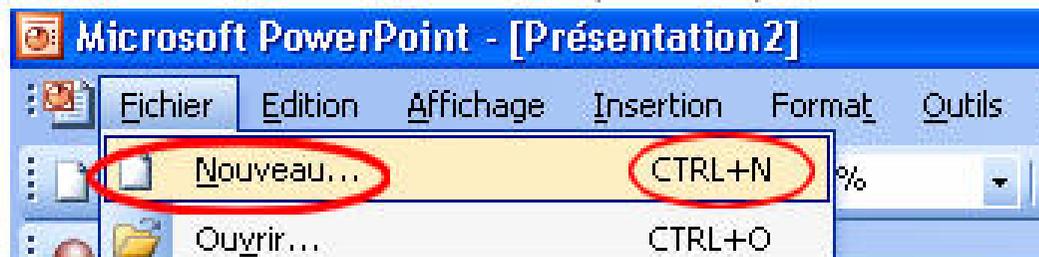


Création d'un diaporama avec PowerPoint

Phase 1: *Création des diapositives*

Pour créer une nouvelle présentation :

- Cliquer sur la commande ***Nouveau*** du menu ***Fichier*** ou sur la combinaison des touches (**ctrl + N**)



- Le ***Volet Office*** affiche ***Nouvelle Présentation***, cliquer sur Nouvelle Présentation

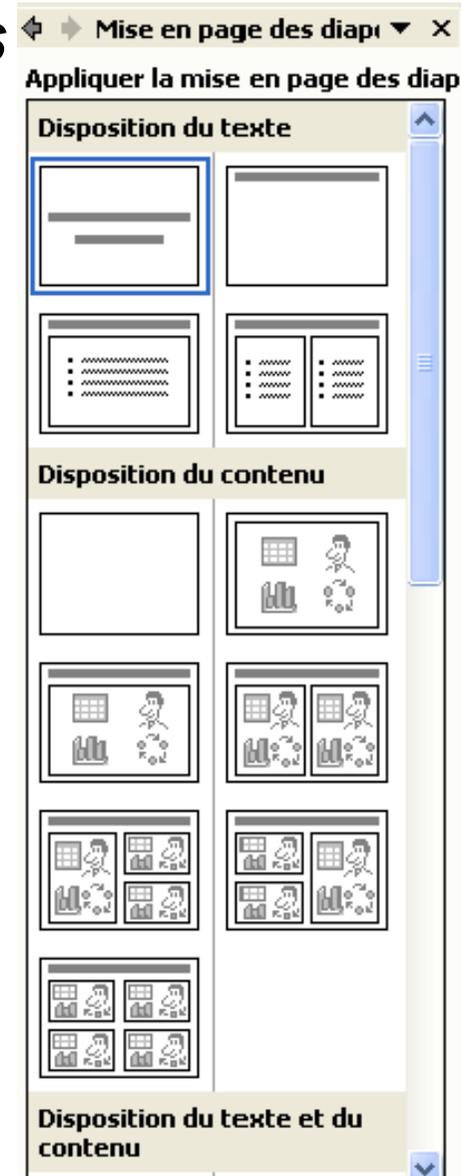


Création d'un diaporama avec PowerPoint

Phase 1: *Création des diapositives*

- Dans le *Volet Office* suivant qui affiche *Mise en page des diapositives*, choisissez une *mise en page automatique* c'est-à-dire une disposition prédéfinie d'éléments graphiques et de zones de texte

Dans PowerPoint, il y a 27 formats préétablis. Ils vous aident à créer une présentation ainsi que d'ajouter les objets dont vous avez besoin.



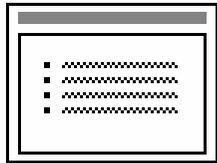


Création d'un diaporama avec PowerPoint

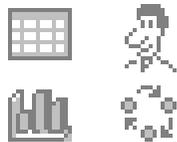
Voici une courte description des éléments que l'on retrouve sur les mises en page des diapositives:



Affiche un titre principal mais laisse le reste de la diapositive vide. Vous pouvez insérer vos propres éléments.



Affiche un titre et du texte.



Permet d'insérer un objet (image, organigramme, image clipart, wordart, tableau ...)



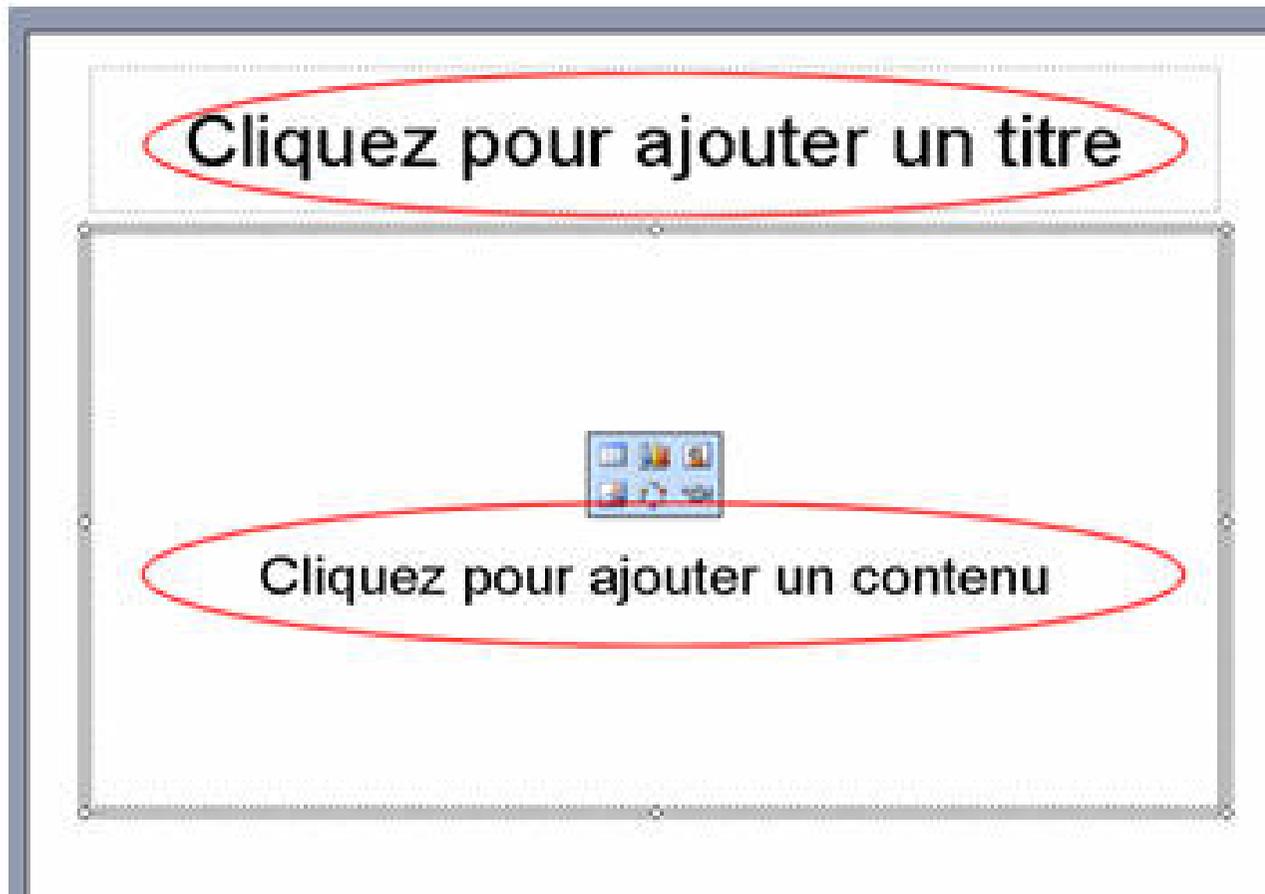
Permet d'insérer du son, une vidéo ou une animation dans votre présentation.



Création d'un diaporama avec PowerPoint

Phase 1: *Création des diapositives*

- Ajoutez un titre, un texte, une image, ...





Création d'un diaporama avec PowerPoint

Phase 1: *Création des diapositives*

- Pour créer une deuxième diapositive on a plusieurs possibilités:
 - Soit par un clic sur l'icône ***Nouvelle diapositive*** de la barre d'outils ***Mise en forme***, puis clic sur la diapositive Vide (du ***Volet Office***).
 - Soit par un clic sur la commande : ***Insertion, Nouvelle diapositive***

Quelque soit la méthode d'insertion utilisée, la deuxième diapositive s'affiche dans le volet central ainsi que la miniature correspondante dans le volet de gauche



Création d'un diaporama avec PowerPoint

Phase 2: *Insertion d'objets sur les diapositives*

Microsoft PowerPoint permet d'insérer des objets de 3 manières:

- Soit par ***Mise en Page*** de diapositives (***Volet Office***),
- Soit par le menu ***Insertion*** (réservé à quelques commandes spécifiques)
- Soit par la barre d'outils ***Dessin***



Création d'un diaporama avec PowerPoint

Phase 2: *Insertion d'objets sur les diapositives*

■ **Insertion d'une image**

Pour insérer une image, et après avoir sélectionné la diapositive sur laquelle on veut Insérer l'image :

- On utilise la commande du menu ***Insertion>Image***,
- Ou on utilise les boutons appropriés pour les images qu'on retrouve sur la zone du ***Volet Office (Mise en page)***.



Image clipart



Image et Photos



Création d'un diaporama avec PowerPoint

Phase 2: *Insertion d'objets sur les diapositives*

■ **Insertion d'un tableau**

Pour insérer un tableau, et après avoir sélectionné la diapositive sur laquelle on veut Insérer le tableau :

- On utilise la commande du menu ***Insertion>Tableau***,
- Ou on utilise le bouton  qu'on retrouve sur la zone du ***Volet Office (Mise en page)***.

Les tableaux de PowerPoint sont similaires à ceux de Word. Ils permettent de créer des présentations, mais pas de calculs.



Création d'un diaporama avec PowerPoint

Phase 2: *Insertion d'objets sur les diapositives*

■ Insertion d'un graphique

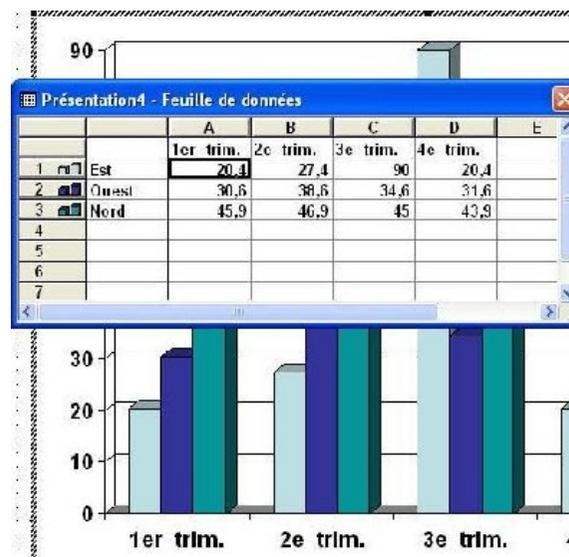
Pour insérer un graphique, et après avoir sélectionné la diapositive sur laquelle on veut Insérer le graphique :

❑ On utilise la commande du menu **Insertion>Graphique**,

❑ Ou on utilise le bouton  qu'on retrouve sur la zone du

Volet Office (Mise en page).

Ici aussi les graphiques de PowerPoint sont similaires à ceux de Word.





Création d'un diaporama avec PowerPoint

Phase 2: *Insertion d'objets sur les diapositives*

■ Insertion d'un diagramme

Pour insérer un diagramme, et après avoir sélectionné la diapositive sur laquelle on veut Insérer le diagramme :

- ❑ On utilise la commande du menu ***Insertion>Diagramme***,
- ❑ Ou on utilise le bouton  qu'on retrouve sur la zone du ***Volet Office (Mise en page)***.





Création d'un diaporama avec PowerPoint

Phase 2: *Insertion d'objets sur les diapositives*

■ **Insertion d'un film ou d'un son**

Pour insérer un film ou un son on utilise la commande **Films et sons** du menu **Insertion**. Cette commande vous permet d'insérer :

- Un film de la bibliothèque multimédia*
- Un film à partir d'un fichier (à partir d'un support de stockage).*
- Un son de la bibliothèque multimédia*
- Un son à partir d'un fichier (à partir d'un support de stockage)*



Création d'un diaporama avec PowerPoint

Phase 3: *Arrière Plan et Modèles de Conception*

Pour améliorer la qualité de nos diapositives, nous allons ajouter un fond qui va nous permettre de modifier l'arrière plan de notre présentation. Deux méthodes sont utilisées:

- *soit à partir d'un fond direct (jeux de couleur, texture, utilisation d'images ...),*
- *soit à partir d'un modèle de conception*

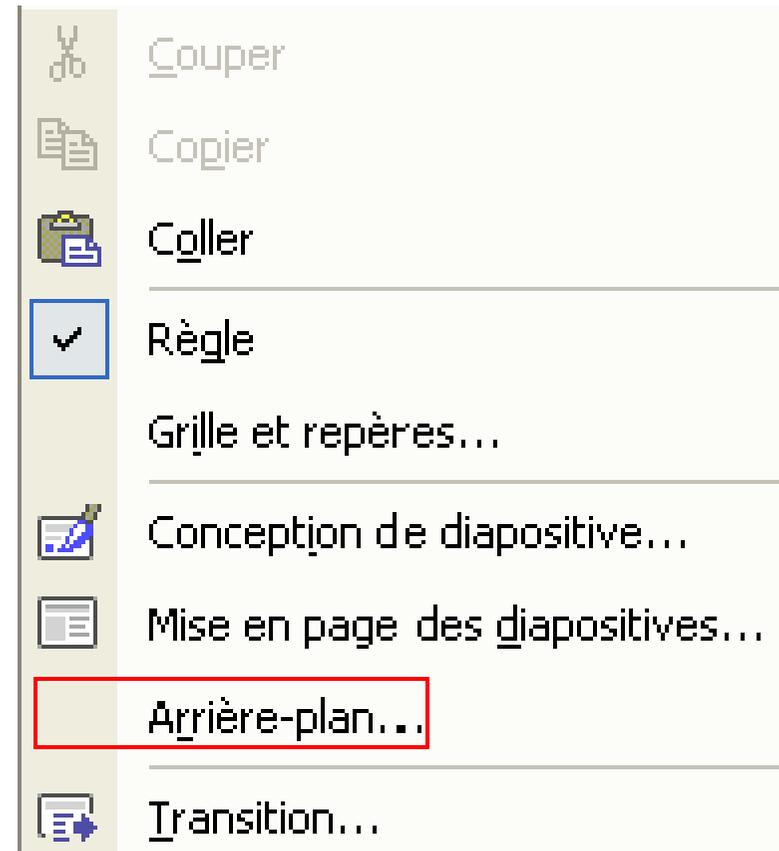


Création d'un diaporama avec PowerPoint

Phase 3: *Arrière Plan et Modèles de Conception*

■ Couleur de Fond, Arrière Plan

On utilise le menu contextuel (clic avec le bouton droit de la souris) au milieu d'une diapositive (hors d'un objet), ou par le menu **Format > Arrière Plan**. La fenêtre **Arrière Plan** va nous permettre de remplacer la couleur d'arrière plan de notre diapositive par une autre au choix.





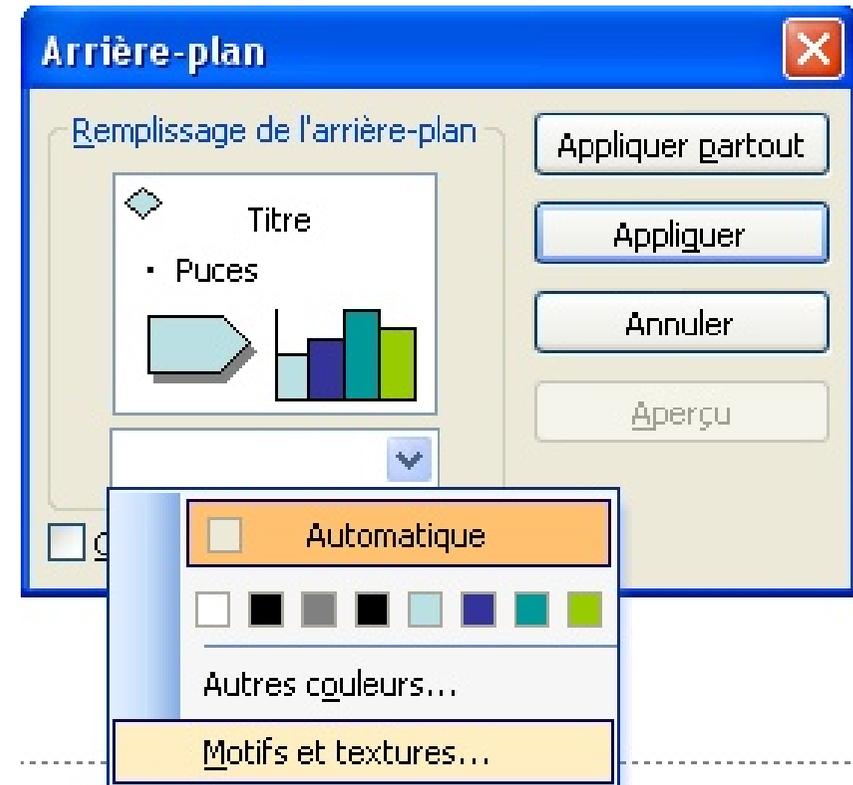
Création d'un diaporama avec PowerPoint

Phase 3: *Arrière Plan et Modèles de Conception*

■ Couleur de Fond, Arrière Plan

Vous pouvez

- ❑ **Appliquer partout**, toutes les diapositives reprendront le nouvel arrière-plan (couleur, image, ...)
- ❑ **Appliquer**, ne s'applique qu'à la diapositive en cours
- ❑ **Aperçu**, permet d'afficher sur la diapositive les modifications effectuées sans enregistrer la modification





Création d'un diaporama avec PowerPoint

Phase 3: *Arrière Plan et Modèles de Conception*

■ **Modèles de Conception**

- Un **Modèle de Conception** PowerPoint est un document avec l'extension **. pot** qui inclut un arrière plan d'origine, un type de fond spécifique et parfois des images (tel qu'un logo) et du texte.
- Pour utiliser un modèle, sélectionnez dans le **Volet Office** ou dans le menu **Format>Conception de diapositive**, la commande **Modèles de conception**



Création d'un diaporama avec PowerPoint

Phase 3: *Arrière Plan et Modèles de Conception*

■ Modèles de Conception

La fenêtre *Modèles de conception* est dissociée en 3 parties:

- *Le modèle utilisé dans cette présentation*
- *Les modèles utilisés récemment*
- *Les modèles disponibles*

Le choix est suffisamment vaste pour permettre à chacun de trouver le meilleur fond possible. Remarquez que selon le modèle de conception choisi, les jeux de couleurs sont différents

Appliquer un modèle de conception





Création d'un diaporama avec PowerPoint

Phase 3: *Arrière Plan et Modèles de Conception*

■ Modèles de Conception

Si vous déplacez le curseur sur la partie gauche du modèle souhaité, le menu suivant apparaît:





Création d'un diaporama avec PowerPoint

Phase 4: *Tri des diapositives*

Ce mode d'affichage vous permet de modifier l'ordre de rangement de vos diapositives.

- Cliquez sur l'icône ***Trieuse de diapositives*** en bas de la fenêtre, à gauche, ou utiliser le menu ***Affichage*** > ***Trieuse de diapositives***
- Dans le mode ***Trieuse*** cliquez sur les diapositives à déplacer puis glissez à l'endroit souhaité



Création d'un diaporama avec PowerPoint

Phase 5: Animations et effet de transition

■ Animations

- Une animation est la capacité d'animer des objets dans une diapositive sans notre intervention directe pendant une présentation.
- Une animation personnalisée nous permet de considérer les objets ou les boîtes de texte dans une diapositive, une par une, et les animer.



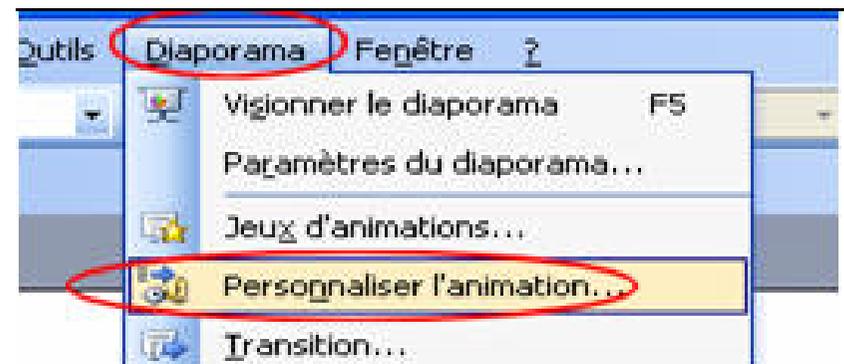
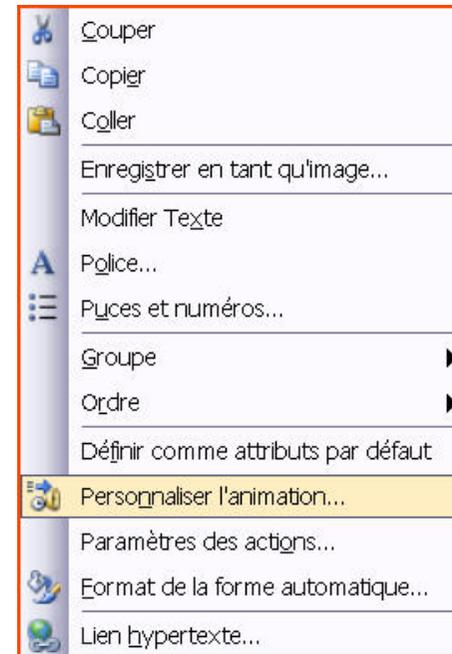
Création d'un diaporama avec PowerPoint

Phase 5: Animations et effet de transition

■ Animations

Pour personnaliser l'animation sur une diapositive :

- Soit vous sélectionnez un objet dans cette diapositive et cliquez sur le bouton droit (de la souris) ensuite cliquez sur **Personnaliser l'animation...**
- Soit sur le menu principal, cliquez **Diaporama > Personnaliser l'animation...**



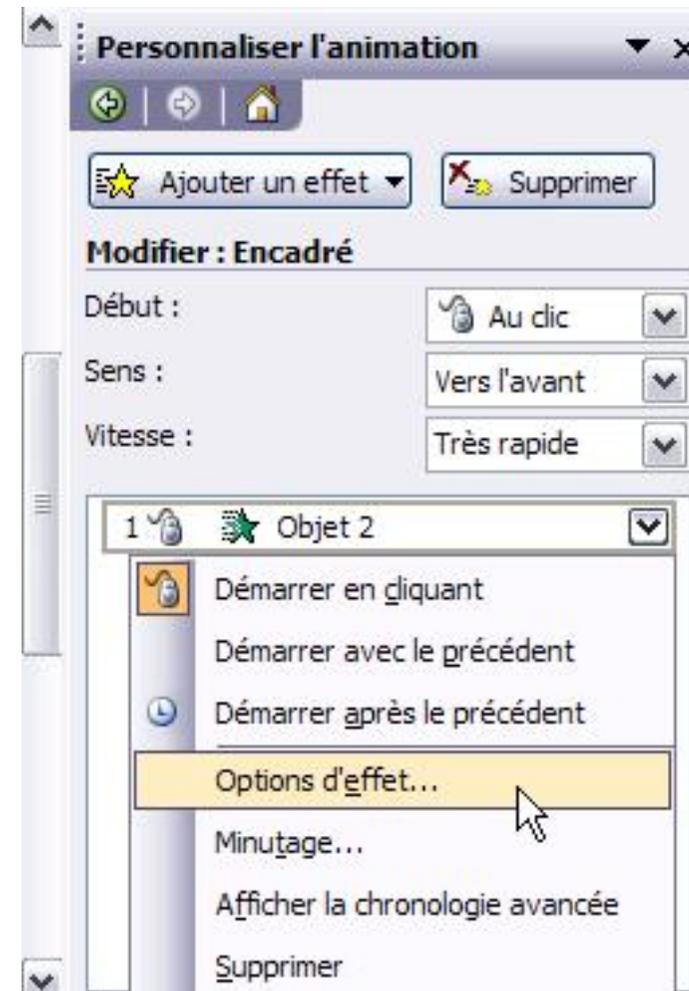


Création d'un diaporama avec PowerPoint

Phase 5: Animations et effet de transition

■ Animations

Ceci va permettre d'afficher le volet **Personnaliser l'animation** à droite de la fenêtre de travail. Ce volet nous permet d'appliquer des animations sur l'objet sélectionné





Création d'un diaporama avec PowerPoint

Phase 5: Animations et effet de transition

■ Animations et sons

- Si vous projetez utiliser le son pendant votre présentation, vous pouvez lui associer quelques effets. PowerPoint dispose de divers effets de son que vous pouvez utiliser pour votre présentation.
- Pour associer un son à une animation, dans l'onglet **Option d'effets...**, sélectionnez un effet son dans la boîte de la section **Son**.



Création d'un diaporama avec PowerPoint

Phase 5: Animations et effet de transition

■ Animations (l'ordre des animations)

Quand vous créez les animations de plusieurs parties d'une diapositive, la liste des animations sur cette diapositive est créée dans la page *Ajouter un effet* de la zone de dialogue *Personnaliser l'animation*. La liste est par accroissement : le premier objet sur lequel vous avez créé l'animation devient le numéro 1, la seconde est le numéro 2. C'est également l'ordre dans lequel ils apparaîtraient dans la diapositive. Si vous voulez un autre ordre, vous pouvez cliquer l'article correspondant à l'animation, et puis cliquez le bouton Haut ou Bas de l'étiquette *Déplacer*



Création d'un diaporama avec PowerPoint

Phase 5: Animations et effet de transition

■ Effet de transition

- La transition des diapositives est le passage d'une diapositive à l'autre, lors de l'affichage du diaporama
- Dans certains cas, vous pouvez vous déplacer d'une diapositive à l'autre dans la présentation automatiquement sans que vous cliquiez sur la diapositive
- Pour activer une transition, on utilise la commande **Diaporama > Transition**





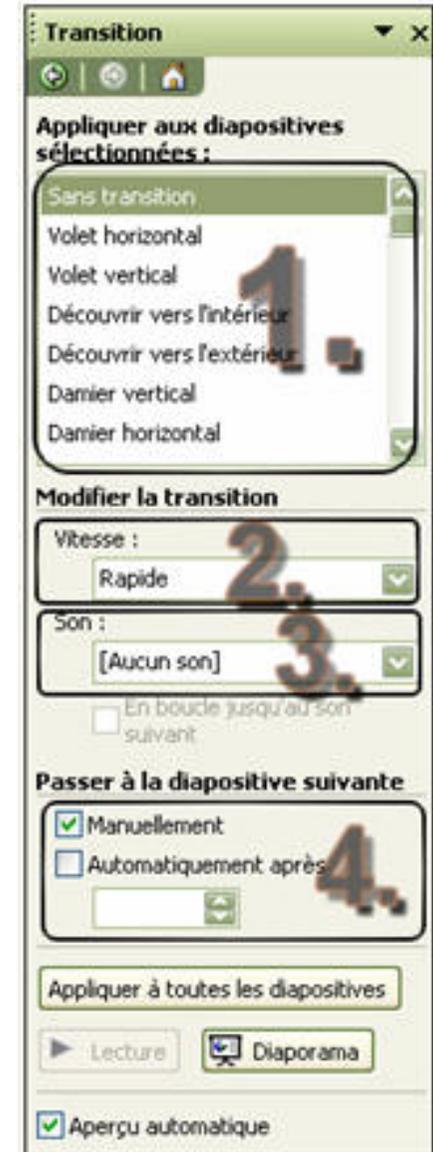
Création d'un diaporama avec PowerPoint

Phase 5: Animations et effet de transition

■ Effet de transition

Une transition peut être paramétrée au niveau de :

- **(1) La forme** : *l'effet de passage (Volet horizontal, damier vertical, dissolution, couper, ...)*
- **(2) La vitesse** : *la vitesse de transition (lente, moyenne, rapide)*
- **(3) Le son** : *émission d'un son l'ors du passage (Acclamation, bombe aspiration, ...)*
- **(4) Le type de passage** : *automatique (après une durée définit) ou manuel (par un clic)*





Création d'un diaporama avec PowerPoint

Phase 5: Animations et effet de transition

■ Effet de transition

Pour afficher du diaporama en continu, sans intervention de l'utilisateur, dans la rubrique ***Passer à la diapositive suivante*** :

- Cocher ***Automatiquement après***
- Sélectionner le temps d'affichage (en secondes) entre chaque diapositive
- Cliquer sur ***Appliquer à toutes les diapositives***

Une transition différente peut être appliquée sur toutes les diapositives. Mais le fait de prendre la même transition pour toutes les diapositives, donne un aspect plus fluide et agréable quand on visionne le diaporama

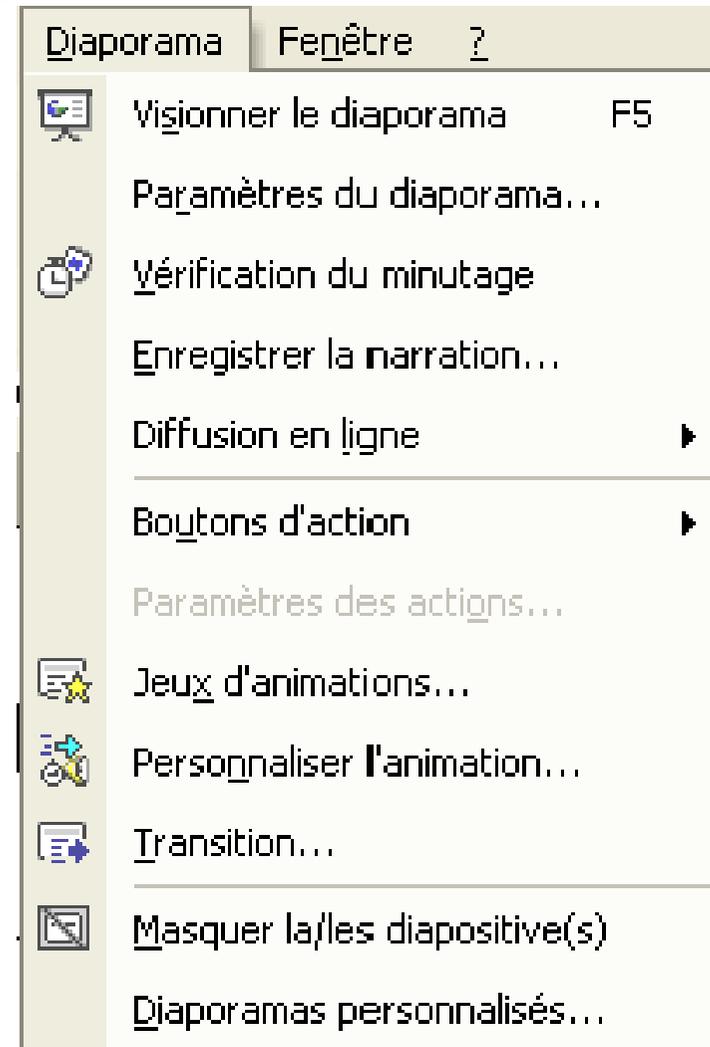


Création d'un diaporama avec PowerPoint

Phase 6: *Présentation du Diaporama*

■ Menu Diaporama

Le menu Diaporama rassemble pratiquement toutes les commandes nécessaires au déroulement de notre diaporama





Création d'un diaporama avec PowerPoint

Phase 6: *Présentation du Diaporama*

■ Visionner le diaporama

Pour démarrer le diaporama :

- Cliquez sur la touche **F5** ou sur la commande **Diaporama > Visionner le diaporama**



- Ou cliquez sur le bouton **Diaporama** en bas à gauche de l'écran 



Création d'un lien hypertexte

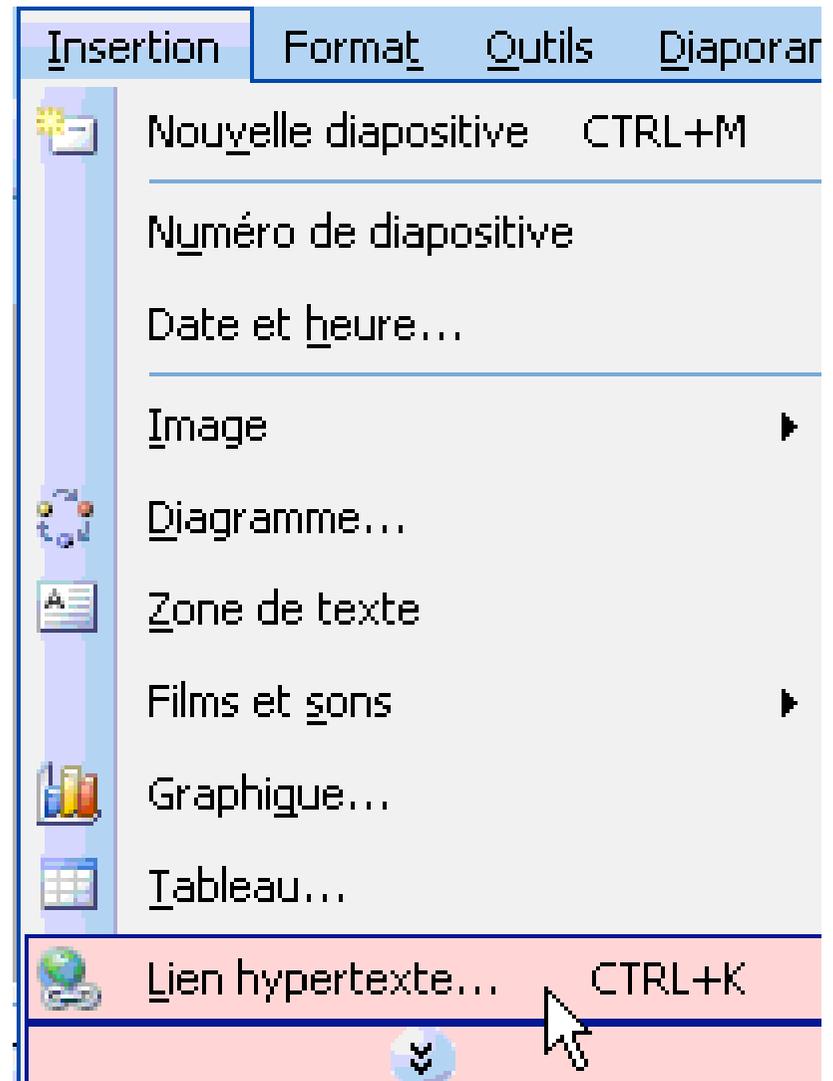
- Dans PowerPoint, un lien hypertexte est une connexion reliant une diapositive à une autre diapositive dans la même présentation, ou dans une autre présentation, à une adresse électronique, à une page Web ou à un fichier
- Vous pouvez créer un lien hypertexte à partir du texte ou d'un objet, tel qu'une image, un graphique, une forme ou un objet WordArt



Création d'un lien hypertexte

Pour créer un **lien hypertexte** :

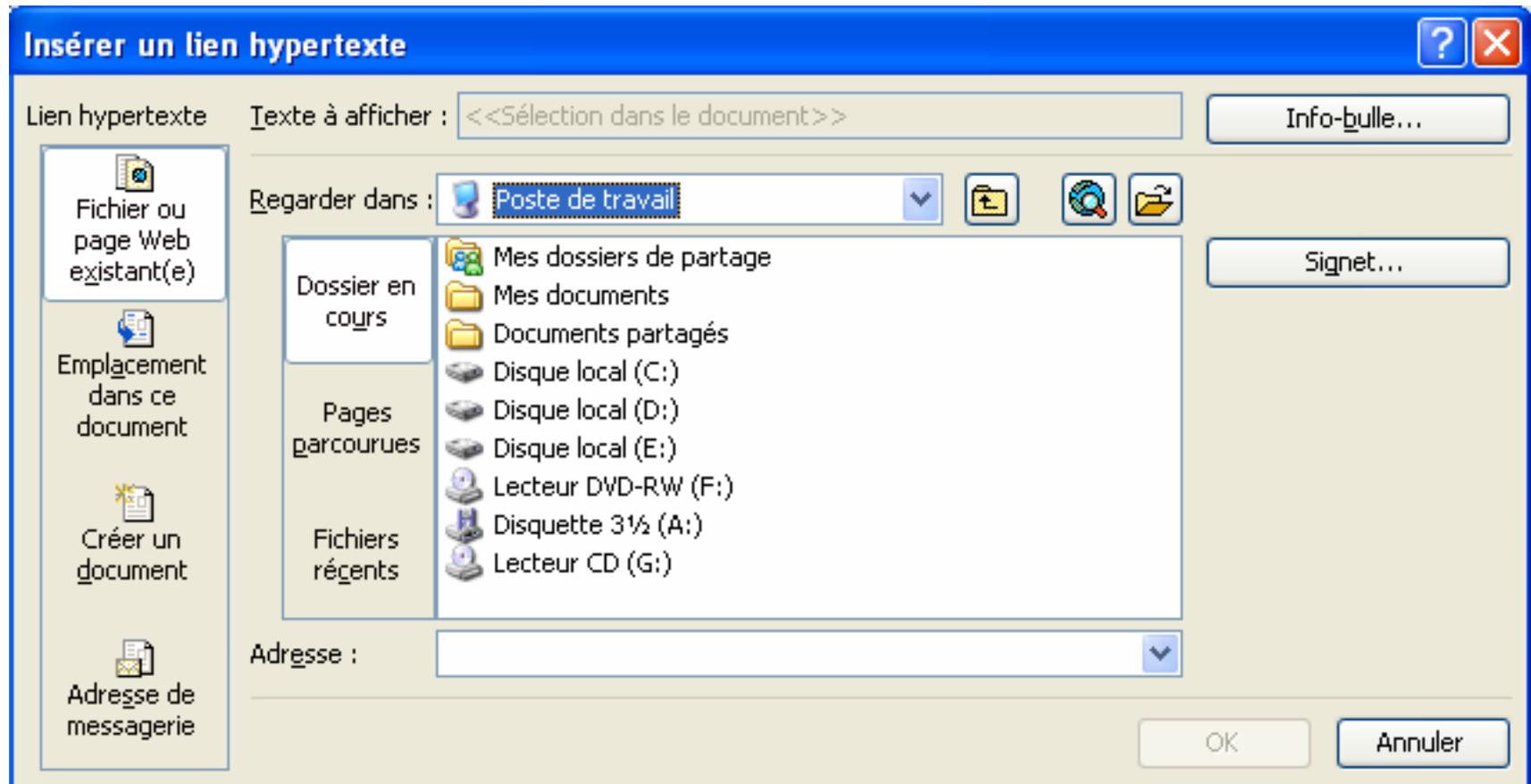
- Sélectionnez le texte ou l'image qui deviendra un lien hypertexte
- Choisissez **Lien hypertexte...** dans le menu **Insertion**





Création d'un lien hypertexte

- Sélectionnez la destination voulue





Cours N°5

MS Office 2003: POWERPOINT





C'est quoi PowerPoint?

- **PowerPoint fait partie de la suite bureautique MS Office éditée par la société Microsoft.**
- **C'est un logiciel classé dans la catégorie des logiciels de Pré – PAO (*Présentation Assistée par Ordinateur*).**
- **Il permet de concevoir, mettre en forme, valoriser les idées qui seront développées lors d'une Présentation (réunion, soutenance, meeting ...).**



C'est quoi PowerPoint?

- **Ce logiciel est capable de gérer des documents aussi variés que :**
 - ❖ **Document A4,**
 - ❖ **Diapositive,**
 - ❖ **Transparents,**
 - ❖ **Rétro-projection directe d'un écran d'ordinateur.**

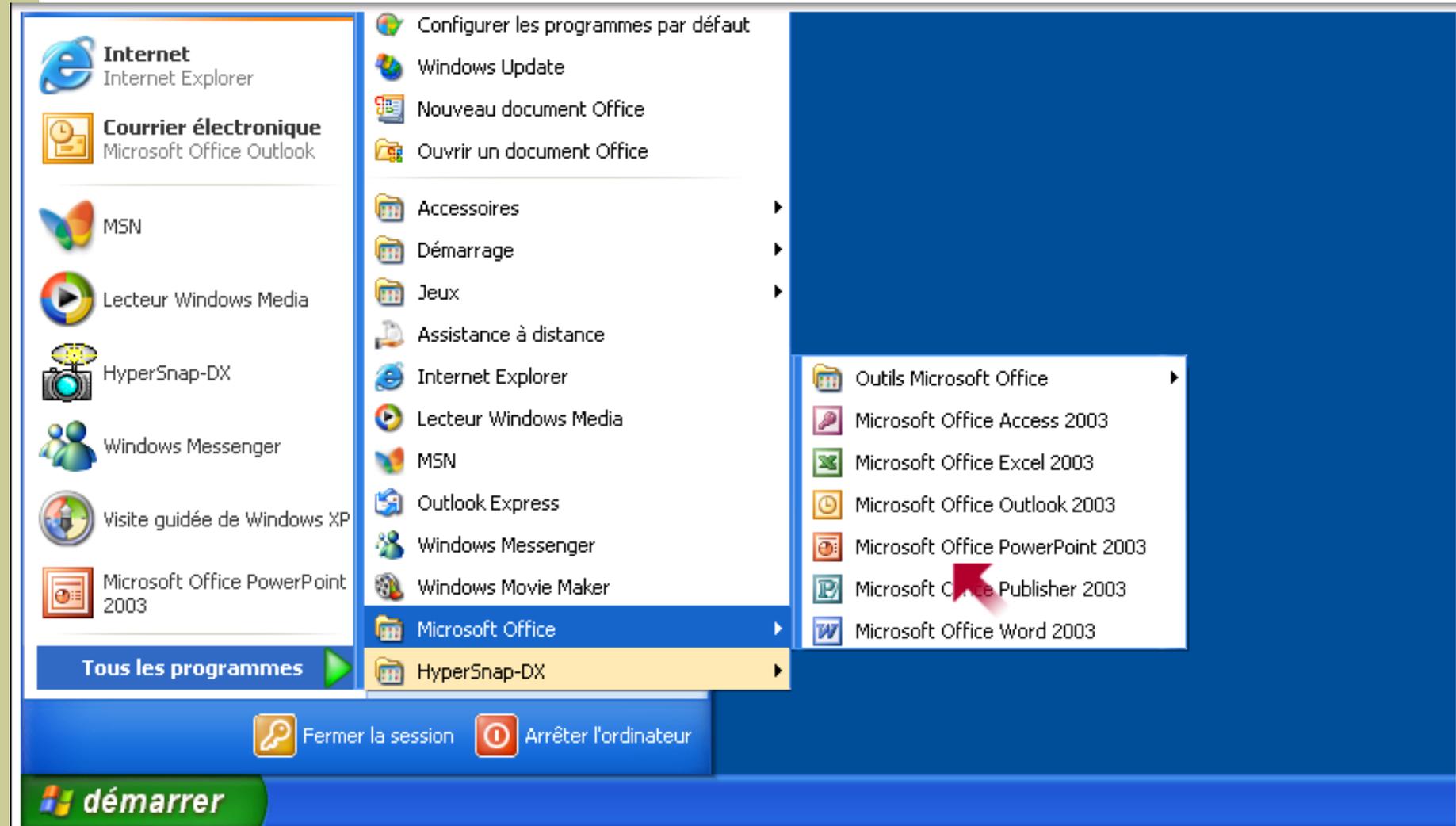


Ce que vous pouvez accomplir avec PowerPoint

- **Créer une présentation sur papier ou la présenter sur ordinateur**
- **Organiser vos idées pour la présentation**
- **Embellir votre présentation pour lui donner une allure "professionnelle"**
- **Ajouter des graphiques, tableaux, organigrammes et autres objets pour améliorer la compréhension des idées que vous voulez présenter**

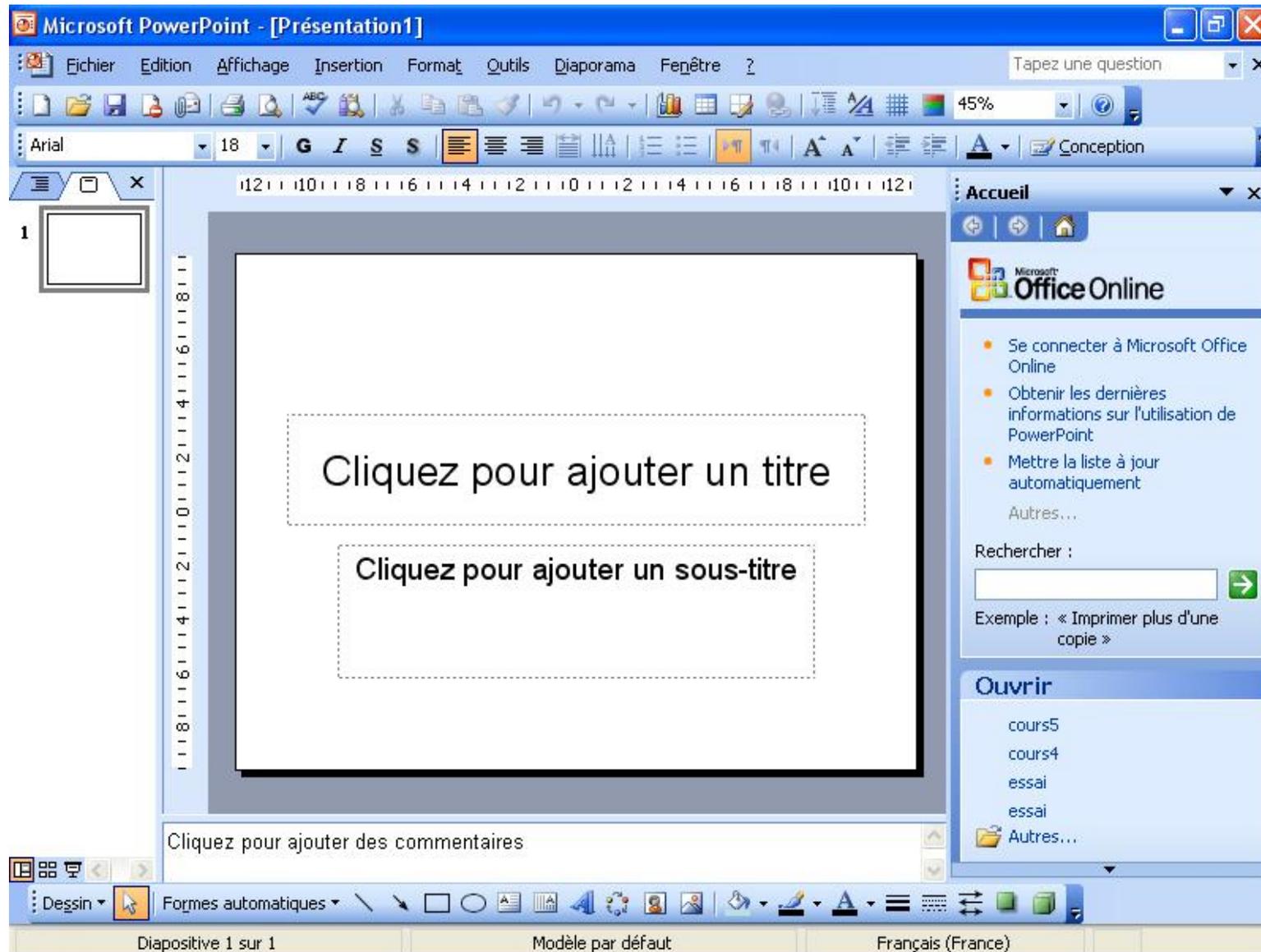


Lancement de l'application PowerPoint





Patientez quelques secondes pendant le chargement de PowerPoint et vous verrez apparaître une fenêtre de ce genre :





L'interface PowerPoint

The screenshot shows the Microsoft PowerPoint 2003 interface. The title bar reads "Microsoft PowerPoint - [Présentation1]". The menu bar includes "Fichier", "Edition", "Affichage", "Insertion", "Format", "Outils", "Diaporama", and "Fenêtre". The toolbar contains various icons for file operations and editing. The main workspace shows a slide with a title box containing "Cliquez pour ajouter un titre" and a subtitle box containing "Cliquez pour ajouter un sous-titre". A status bar at the bottom indicates "Diapositive 1 sur 1", "Modèle par défaut", and "Français (France)".

Callout boxes and their labels:

- Volet Office**: Points to the Office Ribbon (Accueil) on the right side of the interface.
- Volet diapositives en miniatures**: Points to the slide thumbnail pane on the left side.
- La diapositive**: Points to the main slide area.
- Mode d'affichage (Mode plan - Mode trieuse - Mode diaporama)**: Points to the view selection buttons at the bottom left of the interface.



Modes d’Affichage PowerPoint

Pour vous aider à la création de votre présentation, PowerPoint sépare les fonctions du logiciel en 3 "modes" :

- *Mode Normal*
- *Mode Trieuse de Diapositives*
- *Mode Diaporama*

Vous pouvez passer d'un mode à l'autre en utilisant les boutons que vous retrouvez au coin inférieur gauche de l'écran de PowerPoint  ou en utilisant le menu *Affichage* et l'une des 3 premières options du menu



Modes d’Affichage PowerPoint



Mode Normal

- C'est une combinaison des modes Diapositive et Plan
- Il permet la création des pages ou des "diapositives" tout en voyant la structure de votre présentation
- Vous pouvez changer tous les aspects des diapositives (mise en page, les couleurs et la taille du texte, ajouter ou enlever des graphiques, des tableaux, ...)
- Vous pouvez aussi ajouter le texte structuré qui composera votre présentation
- Vous pouvez en tout temps changer l'ordre ou l'importance des idées



Modes d’Affichage PowerPoint



Mode Trieuse de Diapositives

- Ce mode permet d'ajouter des effets de transition (pour le passage d'une diapositive à une autre) à votre présentation si vous songez la présenter en utilisant un ordinateur
- Vous pouvez chronométrer le temps que vous prenez pour donner votre présentation. Vous pourrez ainsi ajuster votre présentation selon le temps qui vous est alloué



Modes d’Affichage PowerPoint



Mode Diaporama

- Ce mode affiche votre présentation à l'écran de l'ordinateur. Même dans ce mode, vous avez accès à des options pour mieux présenter



Définitions

Qu'est-ce qu'une présentation ?

Une **présentation** est une séquence d'images appelées *diapositives*.

Qu'est-ce qu'une diapositive ?

Une **diapositive** est assimilable à **une** page de document contenant du texte, des images, des graphiques, des dessins, du son, des vidéo. Le terme **diapositive** définit un visuel qui est indépendant de l'utilisation qui en sera faite : affichage à l'écran ou impression sur papier ou transparent.



Définitions

Qu'est-ce qu'un diaporama ?

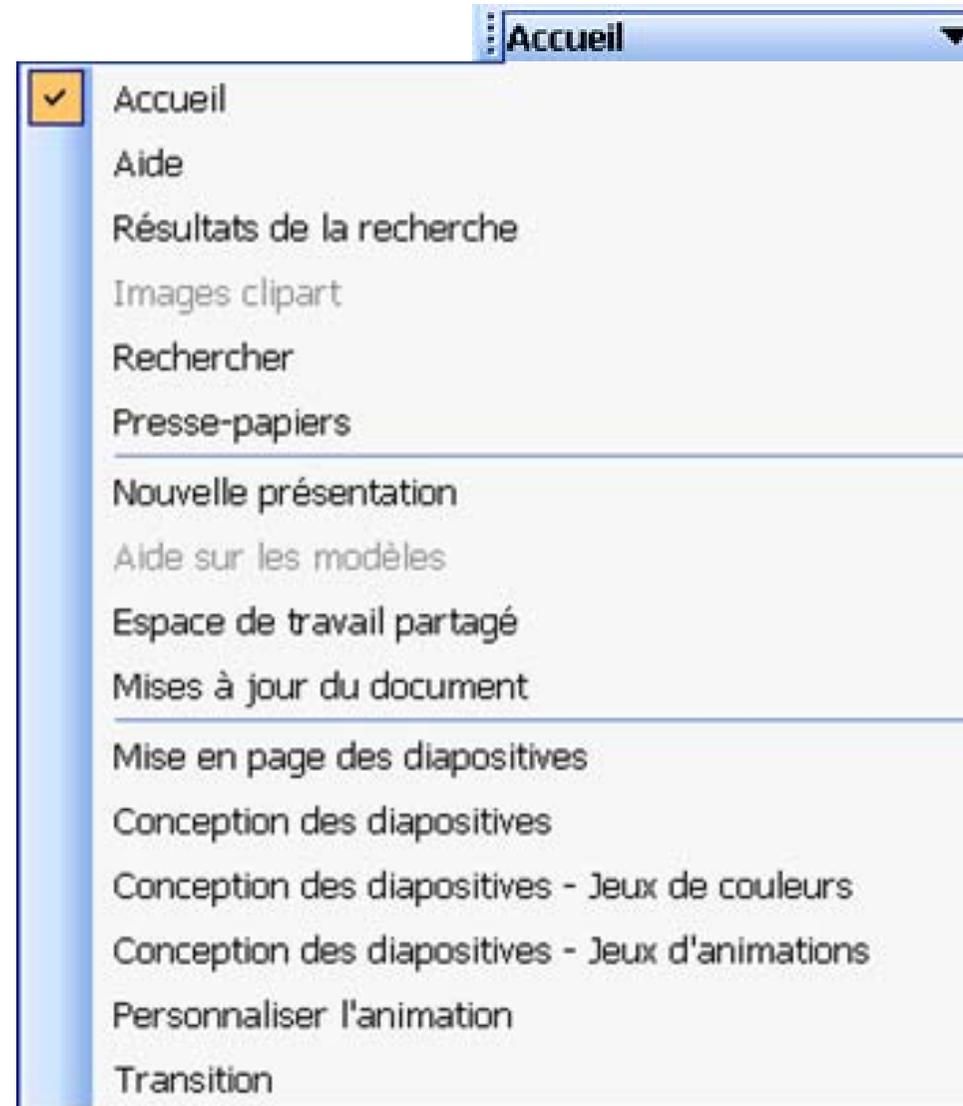
Un **diaporama** est un spectacle de projection de **diapositives** ; par extension on entend par ce terme toute suite d'images ou de documents reliés par des effets et, sur lesquels il est possible de mettre du son.

Toutes les diapositives d'une **présentation** Powerpoint sont contenues dans un seul et même fichier portant l'extension **.ppt**. Si la **présentation** est enregistrée sous forme de **diaporama**, l'extension du fichier sera **.pps**.



Volet Office

Il regroupe les options des assistants ainsi que les options, ou propriétés, des objets placés dans votre présentation



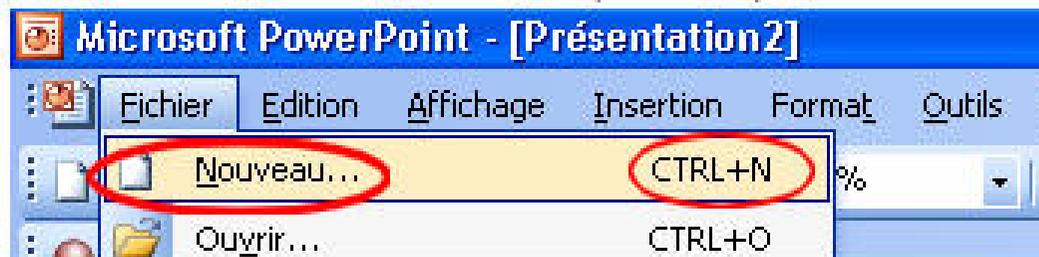


Création d'un diaporama avec PowerPoint

Phase 1: *Création des diapositives*

Pour créer une nouvelle présentation :

- Cliquer sur la commande ***Nouveau*** du menu ***Fichier*** ou sur la combinaison des touches (**ctrl + N**)



- Le ***Volet Office*** affiche ***Nouvelle Présentation***, cliquer sur Nouvelle Présentation

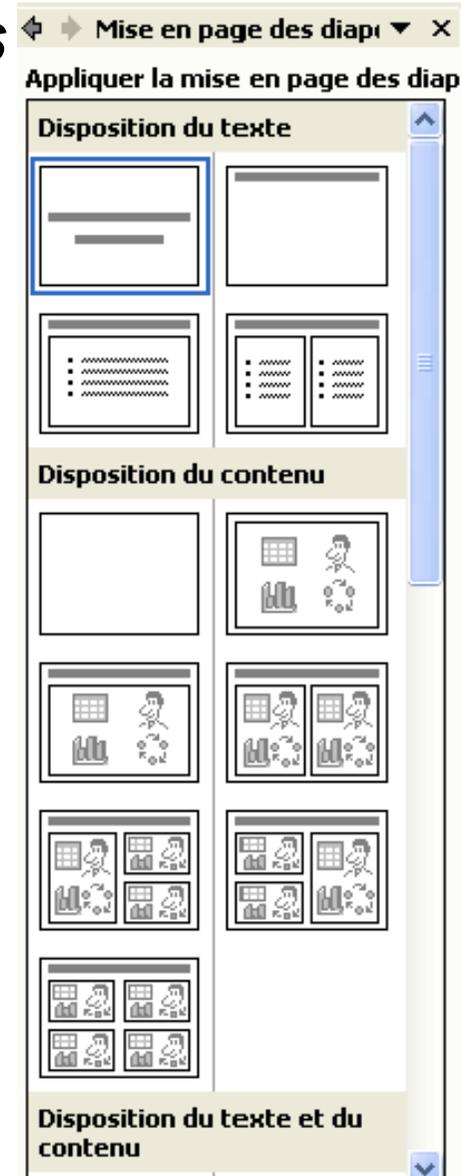


Création d'un diaporama avec PowerPoint

Phase 1: *Création des diapositives*

- Dans le *Volet Office* suivant qui affiche *Mise en page des diapositives*, choisissez une *mise en page automatique* c'est-à-dire une disposition prédéfinie d'éléments graphiques et de zones de texte

Dans PowerPoint, il y a 27 formats préétablis. Ils vous aident à créer une présentation ainsi que d'ajouter les objets dont vous avez besoin.



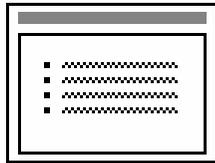


Création d'un diaporama avec PowerPoint

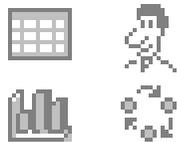
Voici une courte description des éléments que l'on retrouve sur les mises en page des diapositives:



Affiche un titre principal mais laisse le reste de la diapositive vide. Vous pouvez insérer vos propres éléments.



Affiche un titre et du texte.



Permet d'insérer un objet (image, organigramme, image clipart, wordart, tableau ...)



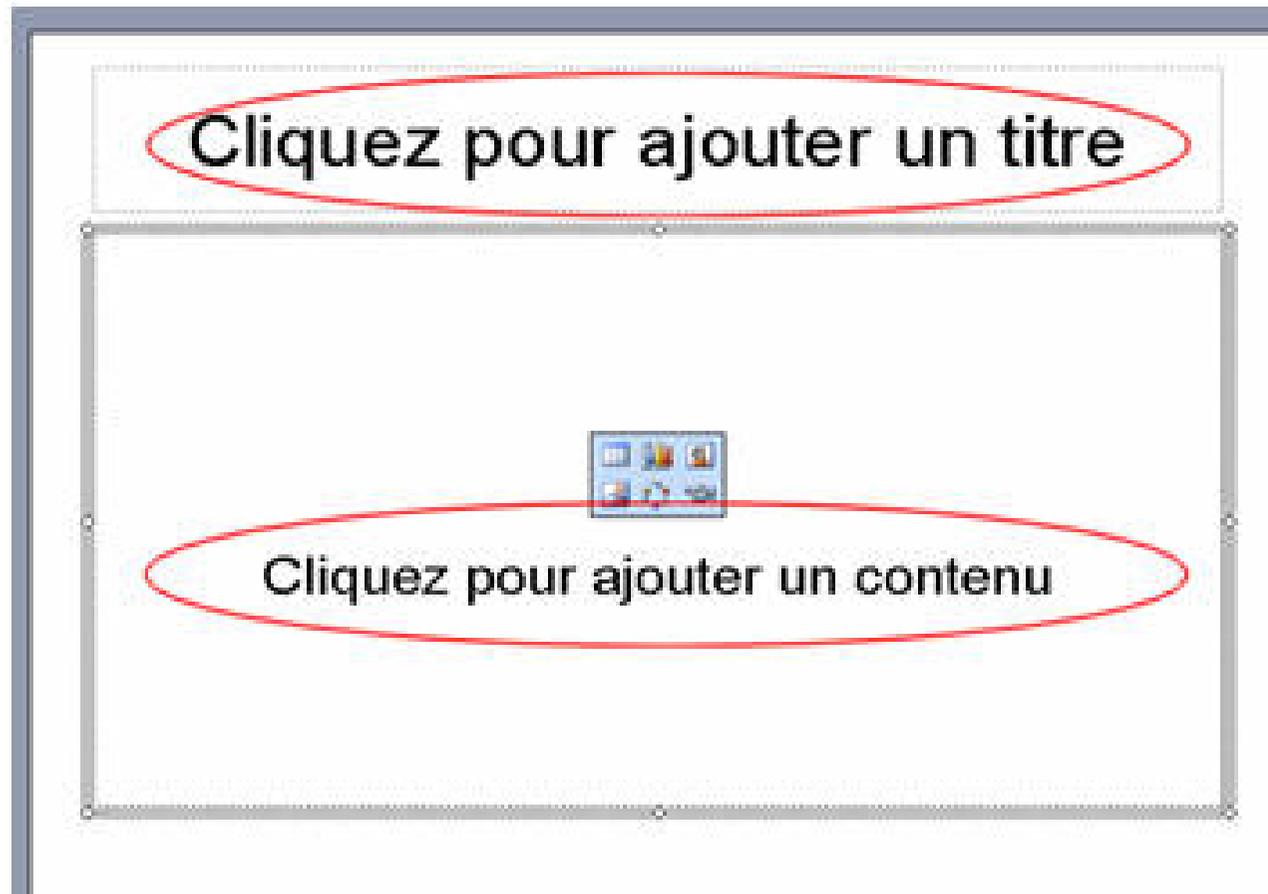
Permet d'insérer du son, une vidéo ou une animation dans votre présentation.



Création d'un diaporama avec PowerPoint

Phase 1: *Création des diapositives*

- Ajoutez un titre, un texte, une image, ...





Création d'un diaporama avec PowerPoint

Phase 1: *Création des diapositives*

- Pour créer une deuxième diapositive on a plusieurs possibilités:
 - Soit par un clic sur l'icône ***Nouvelle diapositive*** de la barre d'outils ***Mise en forme***, puis clic sur la diapositive Vide (du ***Volet Office***).
 - Soit par un clic sur la commande : ***Insertion, Nouvelle diapositive***

Quelque soit la méthode d'insertion utilisée, la deuxième diapositive s'affiche dans le volet central ainsi que la miniature correspondante dans le volet de gauche



Création d'un diaporama avec PowerPoint

Phase 2: *Insertion d'objets sur les diapositives*

Microsoft PowerPoint permet d'insérer des objets de 3 manières:

- Soit par ***Mise en Page*** de diapositives (***Volet Office***),
- Soit par le menu ***Insertion*** (réservé à quelques commandes spécifiques)
- Soit par la barre d'outils ***Dessin***



Création d'un diaporama avec PowerPoint

Phase 2: *Insertion d'objets sur les diapositives*

■ **Insertion d'une image**

Pour insérer une image, et après avoir sélectionné la diapositive sur laquelle on veut Insérer l'image :

- On utilise la commande du menu ***Insertion>Image***,
- Ou on utilise les boutons appropriés pour les images qu'on retrouve sur la zone du ***Volet Office (Mise en page)***.



Image clipart



Image et Photos



Création d'un diaporama avec PowerPoint

Phase 2: *Insertion d'objets sur les diapositives*

■ **Insertion d'un tableau**

Pour insérer un tableau, et après avoir sélectionné la diapositive sur laquelle on veut Insérer le tableau :

- On utilise la commande du menu ***Insertion>Tableau***,
- Ou on utilise le bouton  qu'on retrouve sur la zone du ***Volet Office (Mise en page)***.

Les tableaux de PowerPoint sont similaires à ceux de Word. Ils permettent de créer des présentations, mais pas de calculs.



Création d'un diaporama avec PowerPoint

Phase 2: *Insertion d'objets sur les diapositives*

■ Insertion d'un graphique

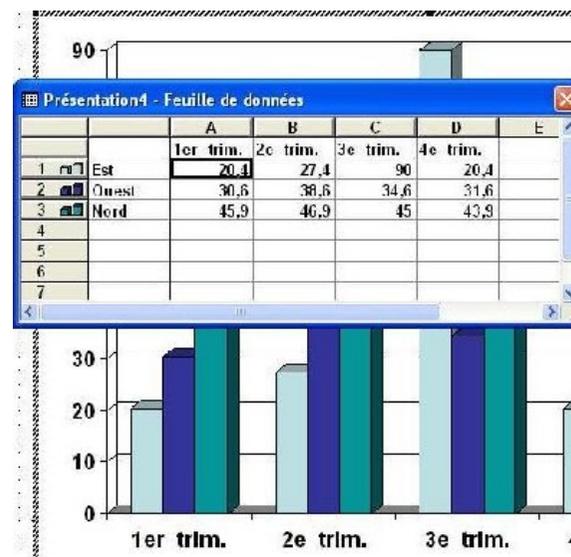
Pour insérer un graphique, et après avoir sélectionné la diapositive sur laquelle on veut Insérer le graphique :

❑ On utilise la commande du menu **Insertion>Graphique**,

❑ Ou on utilise le bouton  qu'on retrouve sur la zone du

Volet Office (Mise en page).

Ici aussi les graphiques de PowerPoint sont similaires à ceux de Word.





Création d'un diaporama avec PowerPoint

Phase 2: *Insertion d'objets sur les diapositives*

■ Insertion d'un diagramme

Pour insérer un diagramme, et après avoir sélectionné la diapositive sur laquelle on veut Insérer le diagramme :

- ❑ On utilise la commande du menu ***Insertion>Diagramme***,
- ❑ Ou on utilise le bouton  qu'on retrouve sur la zone du ***Volet Office (Mise en page)***.





Création d'un diaporama avec PowerPoint

Phase 2: *Insertion d'objets sur les diapositives*

■ **Insertion d'un film ou d'un son**

Pour insérer un film ou un son on utilise la commande ***Films et sons*** du menu ***Insertion***. Cette commande vous permet d'insérer :

- Un film de la bibliothèque multimédia*
- Un film à partir d'un fichier (à partir d'un support de stockage).*
- Un son de la bibliothèque multimédia*
- Un son à partir d'un fichier (à partir d'un support de stockage)*



Création d'un diaporama avec PowerPoint

Phase 3: *Arrière Plan et Modèles de Conception*

Pour améliorer la qualité de nos diapositives, nous allons ajouter un fond qui va nous permettre de modifier l'arrière plan de notre présentation. Deux méthodes sont utilisées:

- *soit à partir d'un fond direct (jeux de couleur, texture, utilisation d'images ...),*
- *soit à partir d'un modèle de conception*

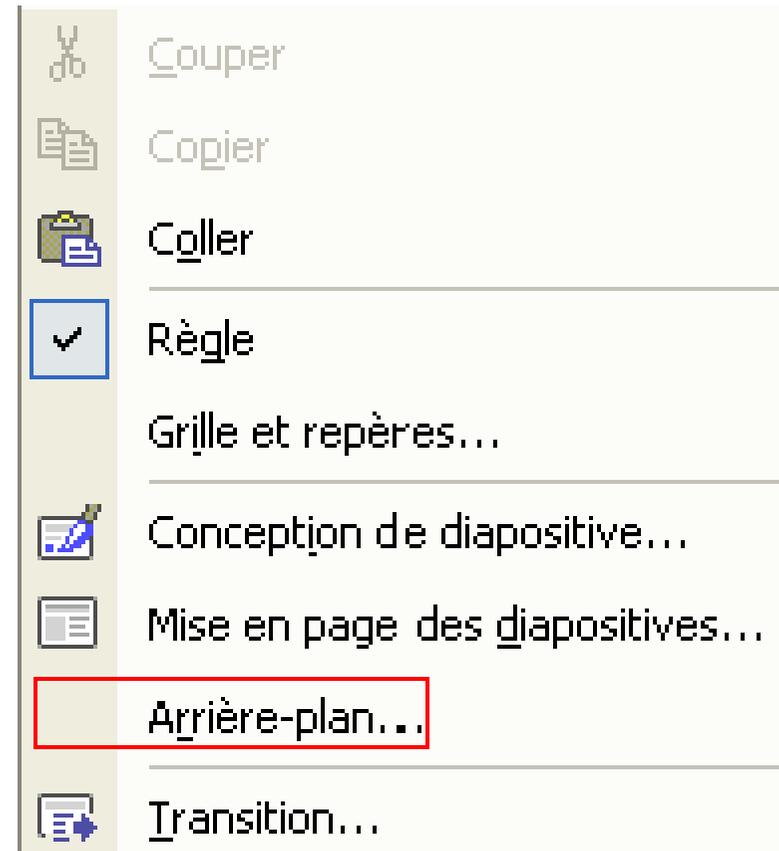


Création d'un diaporama avec PowerPoint

Phase 3: *Arrière Plan et Modèles de Conception*

■ Couleur de Fond, Arrière Plan

On utilise le menu contextuel (clic avec le bouton droit de la souris) au milieu d'une diapositive (hors d'un objet), ou par le menu **Format > Arrière Plan**. La fenêtre **Arrière Plan** va nous permettre de remplacer la couleur d'arrière plan de notre diapositive par une autre au choix.





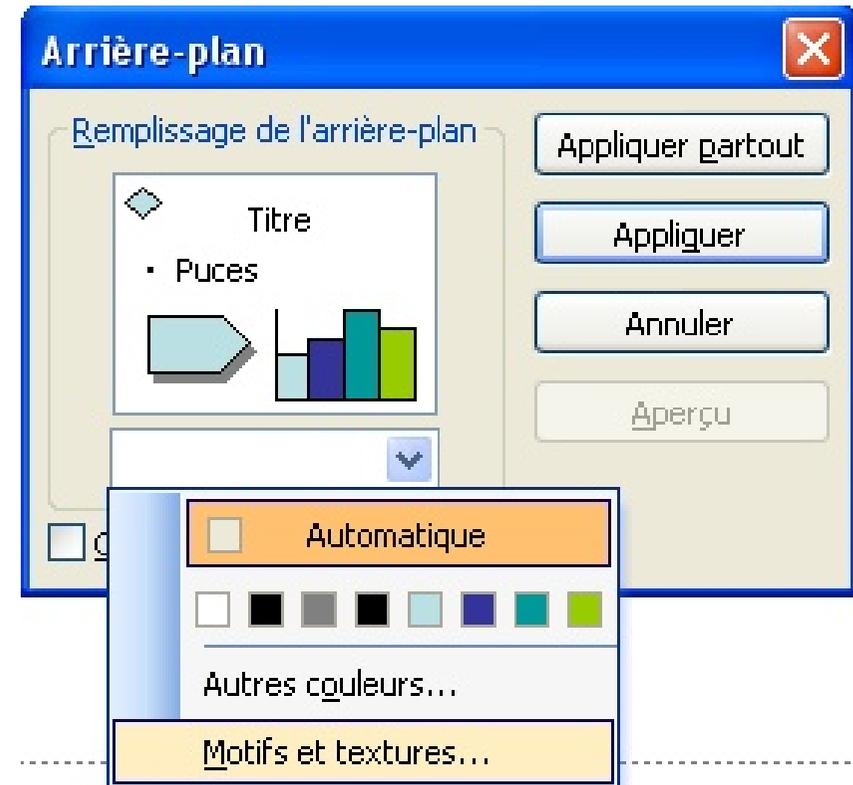
Création d'un diaporama avec PowerPoint

Phase 3: *Arrière Plan et Modèles de Conception*

■ Couleur de Fond, Arrière Plan

Vous pouvez

- ❑ **Appliquer partout**, toutes les diapositives reprendront le nouvel arrière-plan (couleur, image, ...)
- ❑ **Appliquer**, ne s'applique qu'à la diapositive en cours
- ❑ **Aperçu**, permet d'afficher sur la diapositive les modifications effectuées sans enregistrer la modification





Création d'un diaporama avec PowerPoint

Phase 3: *Arrière Plan et Modèles de Conception*

■ **Modèles de Conception**

- Un **Modèle de Conception** PowerPoint est un document avec l'extension **. pot** qui inclut un arrière plan d'origine, un type de fond spécifique et parfois des images (tel qu'un logo) et du texte.
- Pour utiliser un modèle, sélectionnez dans le **Volet Office** ou dans le menu **Format>Conception de diapositive**, la commande **Modèles de conception**



Création d'un diaporama avec PowerPoint

Phase 3: *Arrière Plan et Modèles de Conception*

■ Modèles de Conception

La fenêtre *Modèles de conception* est dissociée en 3 parties:

- *Le modèle utilisé dans cette présentation*
- *Les modèles utilisés récemment*
- *Les modèles disponibles*

Le choix est suffisamment vaste pour permettre à chacun de trouver le meilleur fond possible. Remarquez que selon le modèle de conception choisi, les jeux de couleurs sont différents

Appliquer un modèle de conception





Création d'un diaporama avec PowerPoint

Phase 3: *Arrière Plan et Modèles de Conception*

■ Modèles de Conception

Si vous déplacez le curseur sur la partie gauche du modèle souhaité, le menu suivant apparaît:





Création d'un diaporama avec PowerPoint

Phase 4: *Tri des diapositives*

Ce mode d'affichage vous permet de modifier l'ordre de rangement de vos diapositives.

- Cliquez sur l'icône ***Trieuse de diapositives*** en bas de la fenêtre, à gauche, ou utiliser le menu ***Affichage*** > ***Trieuse de diapositives***
- Dans le mode ***Trieuse*** cliquez sur les diapositives à déplacer puis glissez à l'endroit souhaité



Création d'un diaporama avec PowerPoint

Phase 5: Animations et effet de transition

■ Animations

- Une animation est la capacité d'animer des objets dans une diapositive sans notre intervention directe pendant une présentation.
- Une animation personnalisée nous permet de considérer les objets ou les boîtes de texte dans une diapositive, une par une, et les animer.



Création d'un diaporama avec PowerPoint

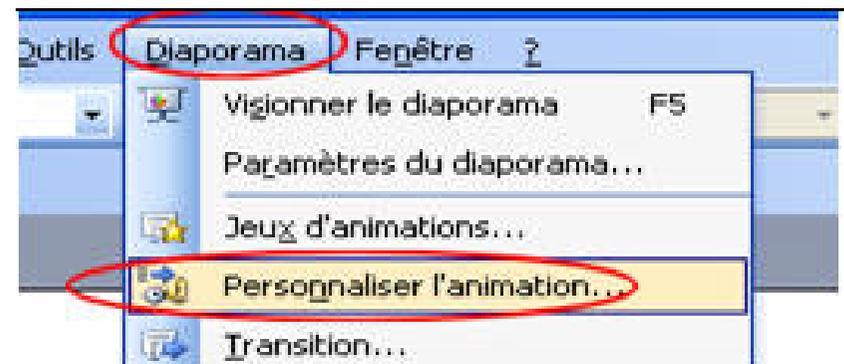
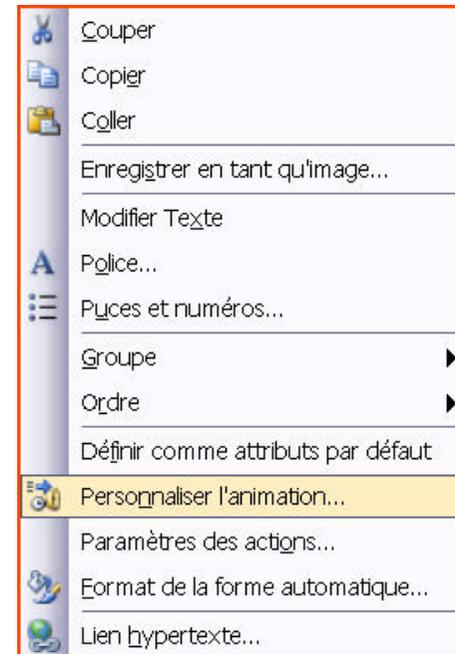
Phase 5: Animations et effet de transition

■ Animations

Pour personnaliser l'animation sur une diapositive :

➤ Soit vous sélectionnez un objet dans cette diapositive et cliquez sur le bouton droit (de la souris) ensuite cliquez sur **Personnaliser l'animation...**

➤ Soit sur le menu principal, cliquez **Diaporama > Personnaliser l'animation...**



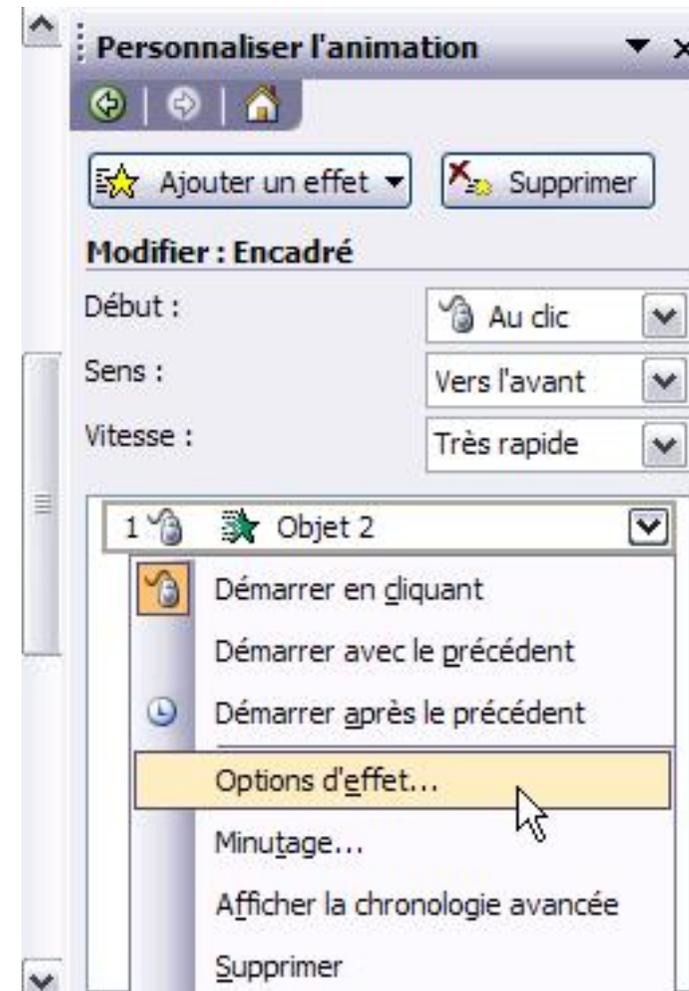


Création d'un diaporama avec PowerPoint

Phase 5: Animations et effet de transition

■ Animations

Ceci va permettre d'afficher le volet **Personnaliser l'animation** à droite de la fenêtre de travail. Ce volet nous permet d'appliquer des animations sur l'objet sélectionné





Création d'un diaporama avec PowerPoint

Phase 5: Animations et effet de transition

■ Animations et sons

- Si vous projetez utiliser le son pendant votre présentation, vous pouvez lui associer quelques effets. PowerPoint dispose de divers effets de son que vous pouvez utiliser pour votre présentation.
- Pour associer un son à une animation, dans l'onglet **Option d'effets...**, sélectionnez un effet son dans la boîte de la section **Son**.



Création d'un diaporama avec PowerPoint

Phase 5: Animations et effet de transition

■ Animations (l'ordre des animations)

Quand vous créez les animations de plusieurs parties d'une diapositive, la liste des animations sur cette diapositive est créée dans la page *Ajouter un effet* de la zone de dialogue *Personnaliser l'animation*. La liste est par accroissement : le premier objet sur lequel vous avez créé l'animation devient le numéro 1, la seconde est le numéro 2. C'est également l'ordre dans lequel ils apparaîtraient dans la diapositive. Si vous voulez un autre ordre, vous pouvez cliquer l'article correspondant à l'animation, et puis cliquez le bouton Haut ou Bas de l'étiquette *Déplacer*



Création d'un diaporama avec PowerPoint

Phase 5: Animations et effet de transition

■ Effet de transition

- La transition des diapositives est le passage d'une diapositive à l'autre, lors de l'affichage du diaporama
- Dans certains cas, vous pouvez vous déplacer d'une diapositive à l'autre dans la présentation automatiquement sans que vous cliquiez sur la diapositive
- Pour activer une transition, on utilise la commande **Diaporama > Transition**





Création d'un diaporama avec PowerPoint

Phase 5: Animations et effet de transition

■ Effet de transition

Une transition peut être paramétrée au niveau de :

- **(1) La forme** : *l'effet de passage (Volet horizontal, damier vertical, dissolution, couper, ...)*
- **(2) La vitesse** : *la vitesse de transition (lente, moyenne, rapide)*
- **(3) Le son** : *émission d'un son l'ors du passage (Acclamation, bombe aspiration, ...)*
- **(4) Le type de passage** : *automatique (après une durée définit) ou manuel (par un clic)*





Création d'un diaporama avec PowerPoint

Phase 5: Animations et effet de transition

■ **Effet de transition**

Pour afficher du diaporama en continu, sans intervention de l'utilisateur, dans la rubrique ***Passer à la diapositive suivante*** :

- Cocher ***Automatiquement après***
- Sélectionner le temps d'affichage (en secondes) entre chaque diapositive
- Cliquer sur ***Appliquer à toutes les diapositives***

Une transition différente peut être appliquée sur toutes les diapositives. Mais le fait de prendre la même transition pour toutes les diapositives, donne un aspect plus fluide et agréable quand on visionne le diaporama

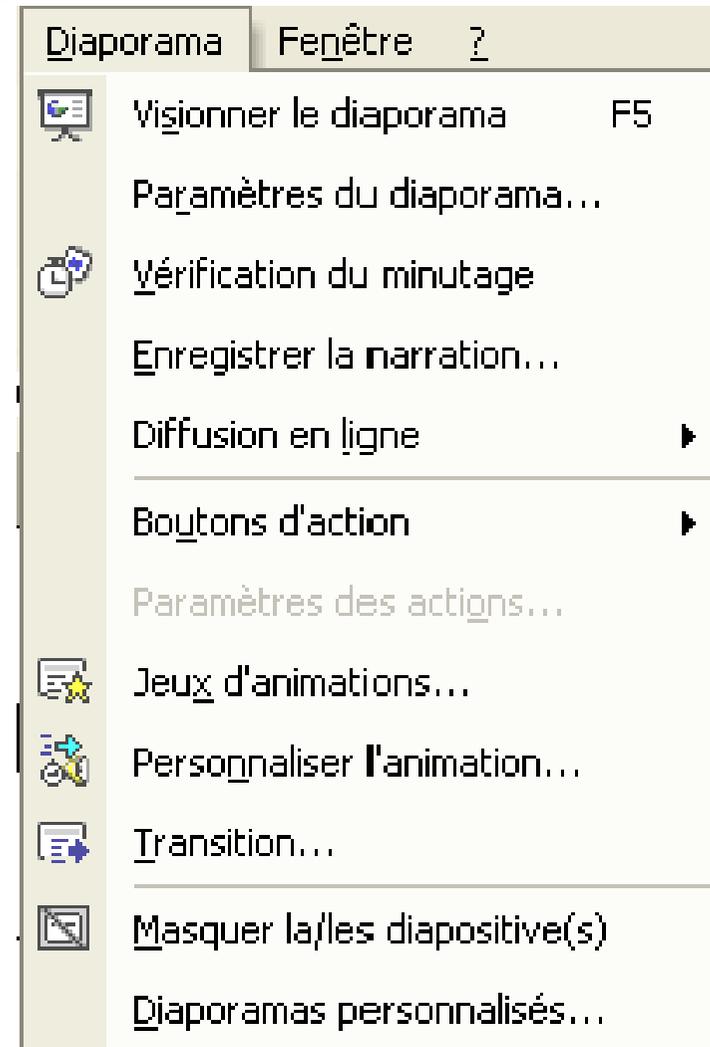


Création d'un diaporama avec PowerPoint

Phase 6: *Présentation du Diaporama*

■ Menu Diaporama

Le menu Diaporama rassemble pratiquement toutes les commandes nécessaires au déroulement de notre diaporama





Création d'un diaporama avec PowerPoint

Phase 6: *Présentation du Diaporama*

■ Visionner le diaporama

Pour démarrer le diaporama :

- Cliquez sur la touche **F5** ou sur la commande **Diaporama > Visionner le diaporama**



- Ou cliquez sur le bouton **Diaporama** en bas à gauche de l'écran 



Création d'un lien hypertexte

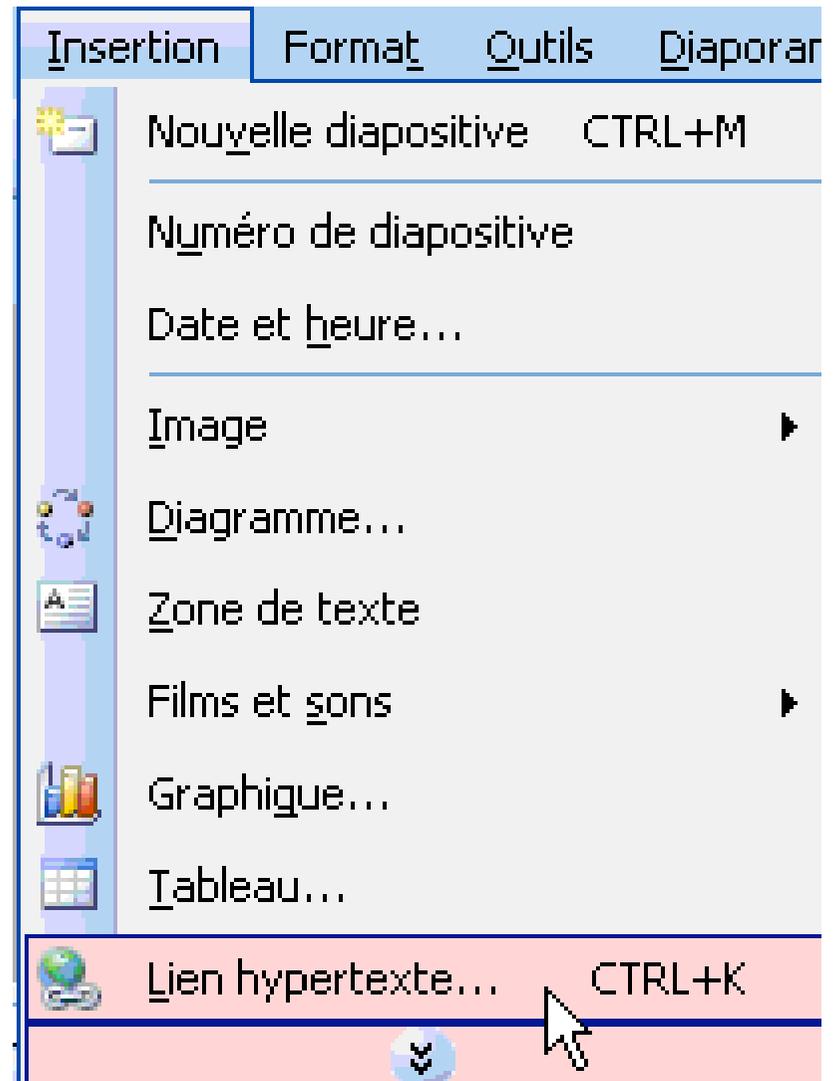
- Dans PowerPoint, un lien hypertexte est une connexion reliant une diapositive à une autre diapositive dans la même présentation, ou dans une autre présentation, à une adresse électronique, à une page Web ou à un fichier
- Vous pouvez créer un lien hypertexte à partir du texte ou d'un objet, tel qu'une image, un graphique, une forme ou un objet WordArt



Création d'un lien hypertexte

Pour créer un **lien hypertexte** :

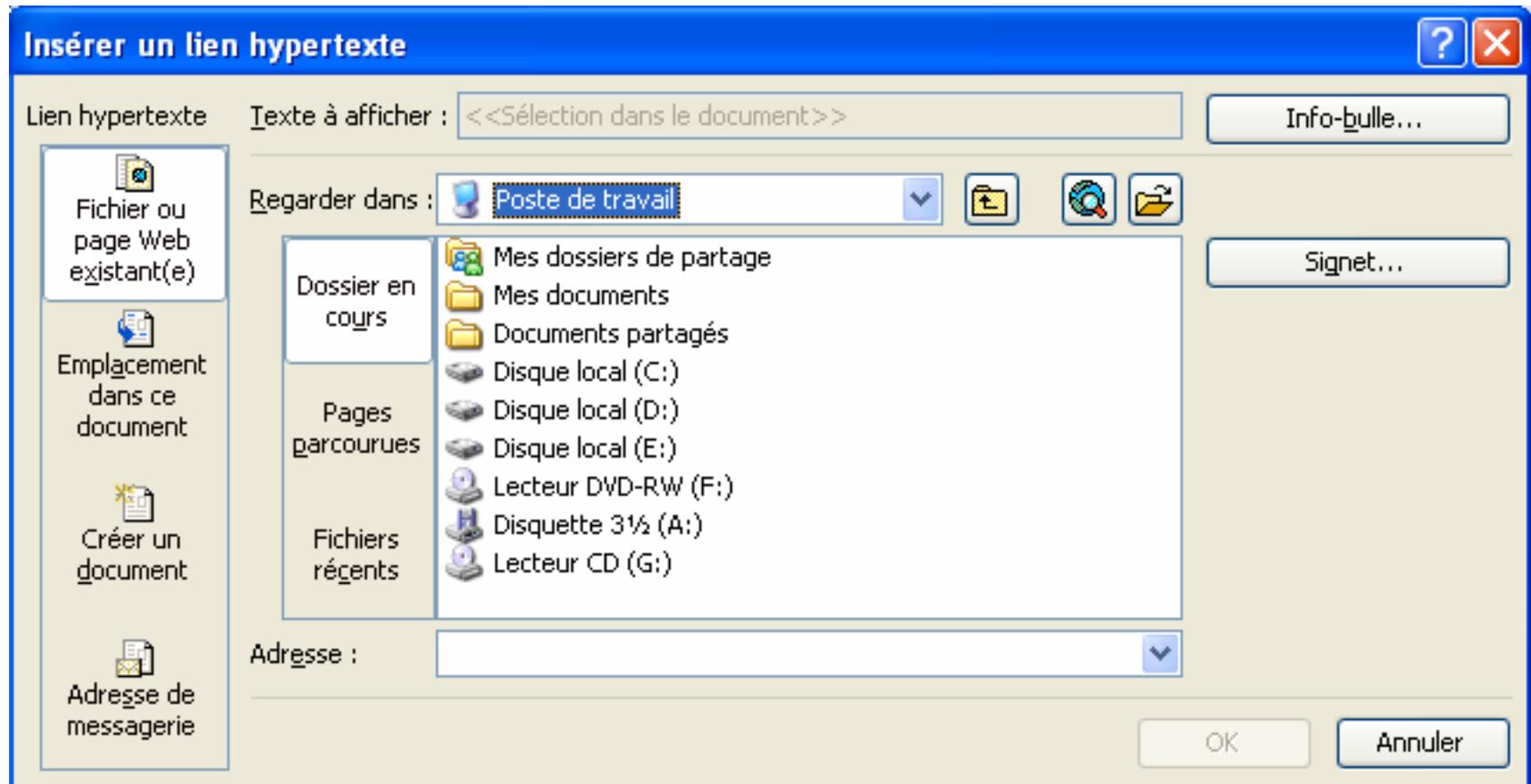
- Sélectionnez le texte ou l'image qui deviendra un lien hypertexte
- Choisissez **Lien hypertexte...** dans le menu **Insertion**





Création d'un lien hypertexte

- Sélectionnez la destination voulue





Cours N°6

Le Traitement des Chaînes de Caractères

Cours élaboré par Mrs RENDAHMANE, BOUFATAH & BRAHMI



2. Représentation des variables chaînes

- En Pascal, les variables chaînes sont nommées au moyen d'un identificateur quelconque
- Les constantes chaînes s'écrivent sous la forme d'une suite de caractères encadrées par des guillemets simple ` `
- Si une apostrophe doit apparaître à l'intérieur de la chaîne, elle doit être doublée

Exemple: `ch := '1''affectation';`



3. Déclaration des variables chaînes

- Les variables chaînes se définissent dans la partie déclarative du programme en précisant leur longueur maximale.
- La déclaration d'une chaîne se fait à l'aide du mot réservé "string" suivie d'une constante entière, entre une paire de crochets, dont la valeur doit être comprise entre 1 et 255.

Exemple:

```
Const lg_max = 30;  
Var X:String[15];  
    Y:String[lg_max];
```



4. Initialisation et affectation d'une chaîne

- On peut affecter à toute variable chaîne une expression ou une constante chaîne de caractères en utilisant le symbole d'affectation traditionnel " := "

Exemple:

Pour les déclarations suivantes:

```
Var X:String[10];  
    Y:String[15];
```

Ces affectations sont correctes:

```
X:= 'Vitesse';  
  
Y:=X;  
  
Y:= 'Temps';  
  
X:=Y;
```



4. Initialisation et affectation (suite)

- Lorsque la variable réceptrice a une longueur maximal inférieure à la longueur de la chaîne qu'on cherche à lui affecter, celle-ci est simplement tronquée par la droite.

Exemple:

Pour les déclarations suivantes:

```
Var x:String[10];  
    y:String[15];
```

```
Y:='Accélération';  
X:=Y;
```

On obtient pour la variable x la chaîne **Accélérati**



5. Comparaison des chaînes

- Les comparaisons de chaînes sont basées, comme les comparaisons des caractères, sur l'ordre des codes ASCII de chacun des caractères qui les constituent.
- L'égalité de deux chaînes a lieu lorsqu'elles ont même longueur courante (pas nécessairement même longueur maximale) et qu'elles sont constituées des mêmes suites de caractères.



5. Comparaison des chaînes (suite)

Exemples:

1. Pour les déclarations suivantes:

```
Var X:String[10];  
    Y:String[15];
```

On obtient

```
X:='Vitesse';  
Y:='Vitesse';  
X:=Y;
```

2. 'vecteur' < 'vitesse'

'Physique1' < 'Physique2'

'bon' < 'bonne'

'12' < '2'



6. Lecture - Écriture des chaînes

A. Instruction de Lecture :

- L'instruction `Readln` permet de lire des chaînes de caractères comme n'importe quelle autre variable d'un autre type.
- En ce qui concerne la manière dont la chaîne est délimitée, le langage Pascal cherche à lire un nombre de caractères correspondant à la longueur maximale, mais il s'interrompt à la rencontre d'une validation par la touche 'Entrer'.
- La lecture doit toujours se faire par `Readln` et non pas par `Read`. (La lecture par `Read` exclut la possibilité de lire convenablement plusieurs chaînes de suite).



6. Lecture - Écriture des chaînes (suite)

A. Instruction de Lecture :

Exemple:

Supposons qu'on veut saisir 2 variables chaînes dont les valeurs sont : **X=Accélération** **Y=Vitesse**

Avec:

```
Var X:String[15];  
    Y:String[10];
```

```
Read(X,Y);
```

Exécution

```
AccélérationVitesse
```



6. Lecture - Écriture des chaînes (suite)

B. Instruction d'Écriture :

- Les instructions "write" et "writeln" permettent d'écrire des chaînes de caractères comme n'importe quelle autre variable d'un autre type.



6. Lecture - Écriture des chaînes (suite)

B. Instruction d'Écriture :

Exemple: Rangement de 2 chaînes de caractères (Par ordre alphabétique)

```
Program rangement;
Uses wincrt;
Var mot,mot1,mot2:String[20];
Begin
  Write(`Donnez un premier mot: `);
  Readln(mot1);
  Write(`Donnez un deuxième mot: `);
  Readln(mot2);
  If mot1 > mot2 Then
    Begin
      mot:=mot1;
      mot1:=mot2;
      mot2:=mot;
    End;
  Writeln(`Voici vos deux mots rangés:`);
  Writeln(mot1,'  ',mot2);
End.
```



7. Lien entre le type Chaîne et le type Caractère

- Le type chaîne et le type caractère sont deux types différents.
- Le type caractère est compatible avec le type chaîne, en revanche le type chaîne n'est pas compatible avec le type caractère.

Exemple:

Pour les déclarations suivantes:

```
Var C:Char;  
    mot:String[1];
```

Ces affectations sont correctes:

```
mot:=C;  
mot:='a';
```

Cette affectation est incorrecte:

```
C:=mot;
```



8. Fonctions et Procédures relatives au type chaîne

A. Fonctions sur les chaînes de caractères :

■ **Fonction Concaténation :**

- Cette opération consiste à juxtaposer deux ou plusieurs chaînes de caractères pour n'en former qu'une seule.
- Elle peut s'obtenir avec l'opérateur "+" ou la fonction "Concat"

Exemple:

Pour les déclarations suivantes:

```
Var A,B,X,Y:String[25];
```

```
A:='Matière';  
B:='Informatique';  
X:=A+' '+B;  
Y:=Concat(A,' ',B);  
Writeln(X);  
Writeln(Y);
```

Exécution

```
Matière Informatique  
Matière Informatique
```

Le résultat d'une concaténation ne peut dépasser 255 caractères. Dans le cas contraire on obtient une erreur à l'exécution.



8. Fonctions et Procédures relatives au type chaîne

(suite)

A. Fonctions sur les chaînes de caractères :

■ **Fonction longueur d'une chaîne (Length) :**

- Cette fonction permet d'obtenir la longueur effective d'une chaîne de caractères (y compris les blancs et les caractères non affichables).
- Le résultat obtenu à partir de cette fonction est de type entier.

Syntaxe: Length(Chaîne)

Exemple:

Pour les déclarations suivantes:

```
mot1 := 'Vitesse';  
mot2 := ' Accélération';  
X := Length(mot1);  
Y := Length(mot2);  
Writeln('X= ', X, ' Y= ', Y);
```

```
Var mot1, mot2: String[15];  
X, Y: Integer;
```

Exécution

X= 7 Y= 13



8. Fonctions et Procédures relatives au type chaîne

(suite)

A. Fonctions sur les chaînes de caractères :

■ **Fonction d'extraction de sous chaîne (Copy) :**

- Cette fonction retourne une chaîne de caractères extraite d'une autre chaîne de caractères .
- La fonction "Copy" permet d'extraire une chaîne d'une longueur donné, à partir d'une position donnée.

Syntaxe: Copy(Chaîne, Position, longueur)

Exemple:

Pour les déclarations suivantes:

```
Var X,Y:String[15];
```

```
X:=`Informatique`;
```

```
Y:=Copy(X,3,6);
```

```
Writeln(`Y= `,Y);
```

Exécution

```
Y= format
```



8. Fonctions et Procédures relatives au type chaîne (suite)

A. Fonctions sur les chaînes de caractères :

■ *Fonction de localisation de sous chaîne (Pos) :*

– Cette fonction permet de situer une sous-chaîne dans une chaîne donnée. Dans le cas où elle s'y trouve, la fonction "Pos" fournit le rang du caractère où commence la sous-chaîne. Dans le cas contraire, elle fournit zéro (0).

Syntaxe: Pos (Sous-Chaîne , Chaîne)

Exemple:

Pour les déclarations suivantes:

```
Var mot1, mot2: String[25];  
X, Y: Integer;
```

```
mot1 := 'Matière Informatique';  
mot2 := 'format';  
X := Pos('Mat', mot1);  
Y := Pos(mot2, mot1);  
Writeln('X= ', X, ' Y= ', Y);
```

Exécution

X= 1 Y= 11



8. Fonctions et Procédures relatives au type chaîne (suite)

B. Procédures sur les chaînes de caractères :

■ **Procédure de suppression de sous chaîne (*Delete*) :**

– Cette procédure permet de supprimer un ou plusieurs caractères d'une chaîne à partir d'une position donnée et avec une longueur donnée.

Syntaxe: `Delete(Chaîne, Position, longueur)`

Exemple:

Pour les déclarations suivantes:

```
Var X:String[25];
```

```
X:='Matière Informatique';
```

```
Delete(X,8,13);
```

```
Writeln('X= ',X);
```

Exécution

X= Matière

Si on cherche à supprimer plus de caractères qu'il n'est possible, il y aura suppression de la partie existante. Si la position indiquée sort des limites de la chaîne, la procédure n'aura aucune action.



8. Fonctions et Procédures relatives au type chaîne (suite)

B. Procédures sur les chaînes de caractères :

■ **Procédure d'insertion de sous chaîne (*Insert*) :**

- Cette procédure permet d'insérer une chaîne de caractères (source) dans une autre chaîne de caractères (destination) à partir d'une position déterminée.

Syntaxe: `Insert(source, destination, Position)`

Exemple:

Pour les déclarations suivantes:

```
Var X,Y:String[25];
```

```
X:= 'Turbo ' ;  
Y:= 'Le Pascal' ;  
Insert(X,Y,4) ;  
Writeln('Y= ',Y) ;
```

Exécution

```
Y= Le Turbo Pascal
```



8. Fonctions et Procédures relatives au type chaîne

(suite)

B. Procédures sur les chaînes de caractères :

■ **Procédure de conversion (*Str*) :**

- Cette procédure effectue la conversion d'un nombre (entier ou réel) en sa représentation sous la forme de chaîne de caractères.

Syntaxe: `Str(nombre, chaîne)`

Exemple:

```
program conversion;
uses wincrt;
var X,Y:string[25];
    S:Real;
Begin
  write('S= ');readln(S);
  STR(S:9:2,X);
  Y:='Num '+X;
  writeln('Y= ',Y);
End.
```

Exécution

S= 10.5

Y= Num 10.50



8. Fonctions et Procédures relatives au type chaîne (suite)

B. Procédures sur les chaînes de caractères :

■ Procédure de conversion (*val*) :

- Cette procédure convertit une chaîne de caractères représentant un nombre en sa valeur numérique.
- Le contenu de la chaîne doit correspondre aux règles d'écriture des nombres. Aucun espace ne doit se trouver en première ou en dernière position.
- Après l'appel de cette procédure, si la conversion a été effectuée, une variable code contient la valeur zéro.
- Dans le cas contraire, cette variable code contient la position du premier caractère de la chaîne qui empêche la conversion, et le contenu de la variable n'est pas défini.



8. Fonctions et Procédures relatives au type chaîne

(suite)

B. Procédures sur les chaînes de caractères :

■ **Procédure de conversion (val) : (suite)**

Syntaxe: val(chaine, nombre, code)

Exemple:

```
program conversion;
uses wincrt;
var  Ch:string[25];
     S,code:integer;

Begin
  write('Ch= ');readln(Ch);
  Val(Ch,S,code);
  writeln('S= ',S);
  writeln('code= ',code);
End.
```

Exécution

```
Ch= 12
S= 12
Code= 0
```



9. Tableau de chaînes de caractères

- Si les chaînes sont des tableaux de caractères, les tableaux de chaînes sont en quelque sorte des tableaux de tableaux.

Exemple:

Pour déclarer un tableau de 10 chaînes de 20 caractères au maximum, on écrit :

```
Var Tab:Array[1..10] of String[20];
```

Nous accédons à chacune des chaînes en utilisant l'élément `Tab[I]` du tableau et nous accédons à n'importe quel caractère `J` de la chaîne `I` à l'aide de `Tab[I,J]`.



Cours N°7

Les Types Définis par l'Utilisateur – Type Ensemble

Cours élaboré par Mrs FENDAH MANE, BOUEATAH & BRAHMI



I. Types Définis par l'Utilisateur

- Jusqu'à maintenant, nous avons vu les types prédéfinis:
 - *Les types simples*: entiers, réels, caractère et booléen
 - *Le type tableau et le type chaînes de caractères*
- Le langage Pascal offre à l'utilisateur la possibilité de définir de nouveaux types de données. C'est ce qu'on appelle ***les types définis par l'utilisateur***.
- Ces nouveaux types permettent à l'utilisateur d'imaginer des traitements à la fois plus performants et plus souples.
- Il existe deux types définis par l'utilisateur:
 - *Le type scalaire par énumération ou le type énuméré*
 - *Le type intervalle*



1. Le Type Énuméré

- Les types énumérés permettent de représenter des valeurs en les énumérant au moyen de leur nom.
- Un type énuméré est un type dont les variables associées n'auront qu'un nombre très limité de valeur (256 au maximum).
- On ne peut alors lui affecter aucune autre valeur que celles prévues dans l'énumération.



1. Le Type Énuméré

A. Définition du type énuméré :

■ On définit une énumération en deux phases:

1) On définit (avant la déclaration des variables) un nouveau type dans lequel on définit l'énumération que l'on souhaite créer (sous forme d'identificateurs), en utilisant la syntaxe suivante:

```
Type <nom_enumeration>=(Val1,Val2,... );
```

2) Une fois l'énumération définie, on déclare des variables, possédant ce type, dans la partie déclarative réservée au variables selon la syntaxe suivante:

```
Var <nom_variable>:<nom_enumeration>;
```



1. Le Type Énuméré

A. Définition du type énuméré :

Exemple :

```
Type jour_semaine = (Samedi,Dimanche,Lundi,Mardi,  
                    Mercredi,Jeudi,Vendredi);  
Var jour : jour_semaine;
```

Dans cet exemple la variable `jour` peut contenir l'une des valeurs prévues dans la définition du type `jour_semaine`, c-à-d: `Samedi`, `Dimanche`, ... , `Vendredi`

Ici il ne s'agit pas de créer une variable chaîne de caractères `jour_semaine` qui peut être égale à l'un des jours de la semaine, mais un nouveau type de variables dont on définit les valeurs possibles.



1. Le Type Énuméré

B. Règles concernant la définition du type énuméré :

- 1) Un identificateur, dans une définition de type énuméré, ne peut pas être un mot réservé.

Exemple :

```
Type Note_Musique = (Do, Re, Mi, Fa, Sol, La, Si);
```

Cette déclaration n'est pas correcte parce que Do est un mot réservé.

- 2) Les constantes dans une déclaration de type par énumération doivent être des identificateurs

Exemple :

```
Type Nombres = (1, 2, 3);
```

Définition fausse

```
Type Nombres = (Un, Deux, Trois);
```

Définition correcte₆



1. Le Type Énuméré

B. Règles concernant la définition du type énuméré :

- 3) Un même identificateur ne peut pas désigner plusieurs choses différentes parce qu'il y a le risque d'ambiguïté.

Exemple :

```
Type Nombres = (Un,Deux,Trois,Quatre,Cinq);  
Numeros = (Un,Deux,Trois);
```

L'identificateur `Un` représente deux entités de type différents.

- 4) Tout identificateur d'une énumération doit être défini avant la déclaration des variable de ce type.

Exemple :

```
Var I,J:Nombres;  
Type Nombres = (Un,Deux,Trois);
```

Définition fausse



1. Le Type Énuméré

C. Propriétés du type énuméré :

- 1) Les types énumérés sont des types ordonnés. L'ordre est défini comme étant celui dans lequel on a énuméré les différentes valeurs du type.

Exemple :

```
Type jour_semaine = (Samedi, Dimanche, Lundi, Mardi,  
                    Mercredi, Jeudi, Vendredi);
```

Samedi < Dimanche < ... < Jeudi < Vendredi

- 2) Les opérateurs relationnels (=, <, >, <=, >=, <>) sont utilisés avec des éléments de même type.
- 3) A chaque valeur énumérée correspond un numéro d'ordre. La première valeur porte le numéro **0**, la seconde le numéro **1** etc.



1. Le Type Énuméré

C. Propriétés du type énuméré :

4) Les fonctions prédéfinies sur les types énumérés sont:

Succ(*expr_type_enumere*) fournit le **successeur** de la valeur donnée par *expr_type_enumere* (erreur si cette expression désigne la dernière valeur énumérée)

Pred(*expr_type_enumere*) fournit le **prédécesseur** de la valeur donnée par *expr_type_enumere* (erreur si cette expression désigne la première valeur énumérée)

Ord(*expr_type_enumere*) fournit le **numéro d'ordre** de la valeur donnée par *expr_type_enumere*

Il existe la fonction réciproque de la fonction **Ord**, c-à-d une fonction qui fait correspondre l'élément du type au rang correspondant. Elle porte simplement le nom du type.



1. Le Type Énuméré

C. Propriétés du type énuméré :

4) Les fonctions prédéfinies sur les types énumérés (suite)

Exemple :

```
Type jour_semaine = ( Samedi, Dimanche, Lundi, Mardi,
                      Mercredi, Jeudi, Vendredi );
```

Succ(Samedi) vaut **Dimanche**

Pred(Jeudi) vaut **Mercredi**

Ord(Samedi) vaut **0**

Ord(Vendredi) vaut **6**

jour_semaine(3) vaut **Mardi**



1. Le Type Énuméré

C. Propriétés du type énuméré :

- 5) Le Pascal n'admet aucune entrée-sortie sur des valeurs d'un type énuméré, c-à-d on ne peut ni lire ni écrire une variable de type énuméré. Néanmoins, il reste possible d'échanger des informations d'un tel type en prévoyant des instructions qui nous permettent de lire ou d'écrire ce type de variables.

Exemple :

Avec la déclaration suivante:

```
Type  Nombres = (Un,Deux,Trois);  
var   Numeros : Nombres;
```

On peut écrire:

```
Case Numeros of  
  Un : writeln('Un');  
  Deux : writeln('Deux');  
  Trois : writeln('Trois');  
End;
```



2. Le Type Intervalle

- Le type Intervalle est un type simple défini par l'utilisateur.
- Un intervalle permet de restreindre le groupe des valeurs d'un type appelé type de base et choisi parmi Integer, Char ou un Type Énuméré (Le type Real est exclu parce qu'il n'est pas de type ordinal).

Exemple :

0..9 est l'intervalle dont le type de base est *integer*

lundi..vendredi est l'intervalle dont le type de base est un
type énuméré

'A'..'Z' est l'intervalle dont le type de base est *char*



2. Le Type Intervalle

A. Définition du type intervalle :

■ On définit un type intervalle en deux phases:

1) On définit (avant la déclaration des variables) un nouveau type intervalle dans lequel on fait état de l'intervalle que l'on souhaite créer, en utilisant la syntaxe suivante:

```
Type <nom_intervalle> = <const_1>..<const_2>;
```

2) Une fois cet intervalle défini, on peut déclarer des variables de ce type selon la syntaxe suivante:

```
Var <nom_variable> : <nom_intervalle>;
```



2. Le Type Intervalle

A. Définition du type intervalle :

■ Remarques:

- ❑ `const_1` et `const_2` sont deux constante de même type tel que: `const_1 <= const_2`
- ❑ La valeur d'une variable de type intervalle ne peut en aucun cas dépasser les bornes de cet intervalle.

Exemple :

```
Type  Majuscule = 'A'..'Z';  
      Chiffre   = 0..20;  
      Nombres  = (Un,Deux,Trois,Quatre);  
      Int_Nombres = Un..Trois;  
Var   Lettre  : Majuscule;  
      X,Y    : Chiffre;  
      Numero : Int_Nombres;
```



2. Le Type Intervalle

B. Propriétés du type intervalle :

- Des variables de type intervalle ont exactement les mêmes propriétés que le type des constantes utilisées dans les bornes de cet intervalle: elles peuvent intervenir dans les mêmes expressions.
- Le seul point de différence est le fait qu'elles ne peuvent pas se voir affecter des valeurs situées en dehors de l'intervalle imposé.



2. Le Type Intervalle

Exemple:

```
Program type_int;  
Uses wincrt;  
Type jour=( Sam,Dim,Lun,Mar,Mer,Jeu,Ven );  
      Nbr_jour=0..6;  
Var   J:jour;  
      nombre:Nbr_jour;  
  
Begin  
  Write('Donnez un nombre de jour (entre 0 et 6): ');  
  Readln(nombre);  
  J:=jour(nombre);  
  If(J=Jeu)or(J=Ven)Then Writeln('C''est un jour de repos')  
  Else Writeln('C''est un jour de la semaine');  
End.
```

Exécution

```
Donnez un nombre de jour (entre 0 et 6): 2  
C'est un jour de la semaine
```

16



3. Type Tableau

- On peut utiliser les tableaux comme étant des types définis par l'utilisateur. Ceci est possible en utilisant la définition suivante:

```
Type <Tableau>=Array[Indice1,Indice2] of <Type_élément>;  
Var <nom_Tableau> : <Tableau>;
```

Exemple :

```
Type Matrice = Array[1..5,1..10] of Real;  
Var A,B,C : Matrice;
```



II. Types Ensembles

- En Pascal, un ensemble est une collection d'objets, comportant des valeurs de même type (maximum 256).
- Comme pour le type intervalle, ces valeurs sont issues de types de base: *Integer*, *Char*, *Boolean*, *Intervalle* ou *un Type Énuméré*.
- Les ensembles sont les mêmes utilisés en mathématique.
- Ils sont donc régis par les mêmes lois classiques de:
 - Réunion
 - Intersection
 - Différence
 - Inclusion
 - Égalité
 - Inégalité
 - Contenance
 - Appartenance



2. Déclaration du Type Ensemble

- On déclare une variable de type ensemble en utilisant la syntaxe suivante:

```
Var <nom_ensemble> : Set of <Type_de_Base>;
```

Exemple :

```
Type Chiffre = 1..10;  
      Nombres = (Un,Deux,Trois,Quatre);  
  
Var   X,Y : Set of Chiffre;  
      Numero : Set of Nombres;  
      Lettre : Set of Char;
```



3. Propriétés du Type Ensemble

- 1) L'affectation de données à une variables déclarée de type ensemble se fait en utilisant les crochets "[]"

Exemple :

Pour la déclaration suivante: `Var Voyelles: Set of Char;`

On peut utiliser l'affectation suivante:

```
Voyelles := [ 'a', 'e', 'i', 'o', 'u', 'y' ];
```

■ Remarques:

- ❑ L'écriture [] indique l'ensemble vide.
- ❑ L'ordre des éléments, dans une affectation ou une comparaison d'ensemble, n'est pas pris en considération puisque les ensembles ne sont pas ordonnés.



3. Propriétés du Type Ensemble

2) Les opérations sur les ensembles sont :

■ Réunion "+" :

Exemple :

```
A := [1, 2, 3];
```

```
B := [3, 4, 5];
```

```
C := A+B;
```

```
C vaut [1, 2, 3, 4, 5]
```

■ Différence "-" :

Exemple :

```
A := [1, 2, 3];
```

```
B := [3, 4, 5];
```

```
C := A-B;
```

```
D := B-A;
```

```
C vaut [1, 2]
```

```
D vaut [4, 5]
```



3. Propriétés du Type Ensemble

2) Les opérations sur les ensembles sont : (suite)

■ **Intersection "*" :**

Exemple :

```
A := [1, 2, 3];
```

```
B := [3, 4, 5];
```

```
C := A * B;
```

```
C vaut [3]
```

■ **Inclusion "<=" :**

Exemple :

```
A := [1, 2];
```

```
B := [1, 2, 3];
```

```
A <= B
```



3. Propriétés du Type Ensemble

2) Les opérations sur les ensembles sont : (suite)

■ **Égalité "=" :**

Exemple :

```
A := [1, 2];
```

```
B := [1, 2];
```

```
A = B
```

■ **Inégalité "<>" :**

Exemple :

```
A := [1, 2];
```

```
B := [3, 4, 5];
```

```
A <> B
```



3. Propriétés du Type Ensemble

2) Les opérations sur les ensembles sont : (suite)

■ **Contenance " \geq " :**

Exemple :

```
A := [1, 2];
```

```
B := [1, 2, 3];
```

```
B  $\geq$  A
```

■ **Appartenance "IN" :**

Exemple :

```
A := [1, 2, 3];
```

```
1 IN A
```

3) Il est impossible d'utiliser les procédures "Read" et "Write" avec les variables de type ensemble.



4. Exemple

```
Program ensemble;  
Uses wincrt;  
Var      Maj : Set of Char;  
         Min : Set of Char;  
         Lettre : Char;  
  
Begin  
  Write('Tapez une Lettre: ');  
  Readln(Lettre);  
  Maj:=['A'..'Z'];  
  Min:=['a'..'z'];  
  If (Lettre IN Maj) Then Writeln('C''est une MAJUSCULE')  
  Else If (Lettre IN Min) Then Writeln('C''est une minuscule')  
  Else Writeln('Ce n''est pas une lettre');  
  
End.
```

Exécution

```
Tapez une Lettre: A  
C'est une MAJUSCULE
```

25