

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
École Supérieure d'Économie d'Oran



Polycopié pédagogique

Cours et exercices corrigés en Informatique

destinés aux étudiants de la première année des classes préparatoires

Élaboré par Dr Souad MADOURI

Année universitaire 2022 / 2023

Table des matières

CHAPITRE 1 : Les bases de l'informatique

1. Concepts de base de l'informatique, de l'information Et de l'ordinateur.....	7
1.1 Introduction à l'informatique.....	7
- définition.....	7
- utilisation de l'Informatique.....	7
1.2 Introduction à l'information.....	7
-définition.....	7
1.3 Ordinateurs : - définition.....	8
2. Notions sur le fonctionnement d'un pc.	9
a/Le matériel.....	9
b/Le logiciel (software)	13
3. Les réseaux et sécurité informatiques.....	13

CHAPITRE 2 : Le système de numération

1. Introduction	16
2. Le transcodage : les différentes conversions.....	17
3. Le transcodage d'un texte en langage binaire	19

CHAPITRE 3 : Bases de l'algorithmique

1. Définition du mot Algorithme	22
2. La structure d'un algorithme	22
3. Constante.....	22
4. Variable	23
5. Les types de variable.....	23
6. Les instructions.....	23
A/ Instruction d'affectation	23
B/ Instruction de lecture	24
C/ Instruction d'écriture	24
D/ Les structures de contrôles.....	26
- Introduction	26
- Le traitement séquentiel	26
- Le traitement alternatif	26

1/ Alternatif simple	26
2/ Alternatif composé	27
3/ Imbrications de conditions	28
E/ Le traitement répétitif (iterative).....	30
1/La boucle pour	30
2/La boucle tant que	30
3/La boucle répéter	31

CHAPITRE 4 : Présentation Turbo-Pascal Windows

1. Un programme	33
2. Structure générale d'un programme PASCAL	33
3. La déclaration des constantes.....	33
4. La déclaration des variables.....	33
5. Les types de donnée de base dans Pascal.....	34
6. Instructions élémentaires	34
7. Affectation	34
• Les instructions READ/WRITE	34
-L'instruction write().....	34
-L'instruction read()	35
• Les instructions conditionnelles.....	35
• Structure conditionnelle avec alternative.....	36
• Imbrications de conditions	36
-Les boucles.....	37
For.....do	37
Whiledo	38
Repeat...until	39

CHAPITRE 5 : Les structures de Données Complexes

1. Les Tableaux à une dimension	40
1.1. Définition d'un tableau.....	40
1.2. La déclaration d'un tableau à une dimension	40
1.3. Manipulation d'un tableau	40
1.4. La lecture d'un tableau à une dimension	41
1.5. L'affichage d'un tableau à une dimension	41

2. Les Tableaux à deux dimensions	43
2.1. La déclaration d'un tableau à deux dimensions	43
2.2. La lecture d'un tableau à deux dimensions	43
2.3. L'affichage d'un tableau à deux dimensions	43
Examen avec correction.....	46
Ressources bibliographiques.....	55

Liste des figures

N°	Titre	Page
1	Schéma du traitement automatique de l'information	7
2	Schéma simplifié d'un ordinateur	8
3	Schéma général de l'unité central	9
4	Le matériel : schéma général de l'ordinateur	9
5	le processeur	10
6	la mémoire	10
7	le disque dur	11
8	La carte mère	11
9	Carte graphique	12
10	Les réseaux et sécurité informatiques	14

Liste des tableaux

N°	Titre	Page
1	Les différents systèmes de numération	16
2	Le code ASCII	19

Introduction

Le présent polycopié représente un cours d'informatique adressé aux étudiants de la première année école supérieur d'économie. Ce polycopié est élaboré dans le but de permettre à l'étudiant d'identifier, de comprendre et de maîtriser les outils de base et les principaux concepts utilisés dans le domaine de l'informatique.

Il est constitué de cinq chapitres :

Les éléments de logiques ; Les Bases de L'Informatique, Le système de numération , afin de maîtriser les concepts de base qui permet à l'étudiant de bien profiter des avantages et des qualités offertes par les nouvelles technologies dans leur cursus.

Le quatrième chapitre Base de l'Algorithmique, et Enfin, dernier chapitre Les structures de Données Complexes à pour objectif de continuer les principes de base de notions d'algorithmique et de bien maîtriser les structures de données complexes, apprendre la programmation.

Les Bases de L'Informatique

1. Concepts de base de l'informatique, de l'information Et de l'ordinateur

1.1. Introduction à l'informatique:

- Définition:

➤ traitement automatique de l'information à l'aide de programmes mis en œuvre sur ordinateur.



Figure 1: Schéma du traitement automatique de l'information

https://sites.google.com/site/otmani1981/_/rsrc/1470336890657/initiation-a-l-informatique

-Utilisation de l'informatique:

Il existe de nombreux domaines d'utilisation de l'outil informatique:

- Le domaine industriel;
- Le domaine banque/finance/comptabilité;
- Les domaines de la téléphonie, des communications et des médias;
- Le domaine scientifique et l'énergie ;
- Le domaine des administrations publiques et entreprises (publiques ou privées)

1.2. Introduction à l'information:

-Définition:

- Information = données + sens

- Une information
-
- Le diagramme montre un rectangle noir avec trois flèches rouges pointant vers des étiquettes à droite. La flèche supérieure pointe vers 'Image (image)', la flèche du milieu vers 'Texte (texte)' et la flèche inférieure vers 'Entendue (sons)'.

- Pour traiter l'information, elle doit être **codée (numérisée)**

1.3. Ordinateur:

- Un ordinateur est une machine qui permet de travailler avec des informations (nombres, mots, images, sons) appelées données.
- Les ordinateurs peuvent traiter très vite de grandes quantités de données, les stocker et les afficher.
- Machine électronique possédant une certaine architecture matérielle (hardware) lui permettant d'exécuter des programmes, ou logiciels (software).

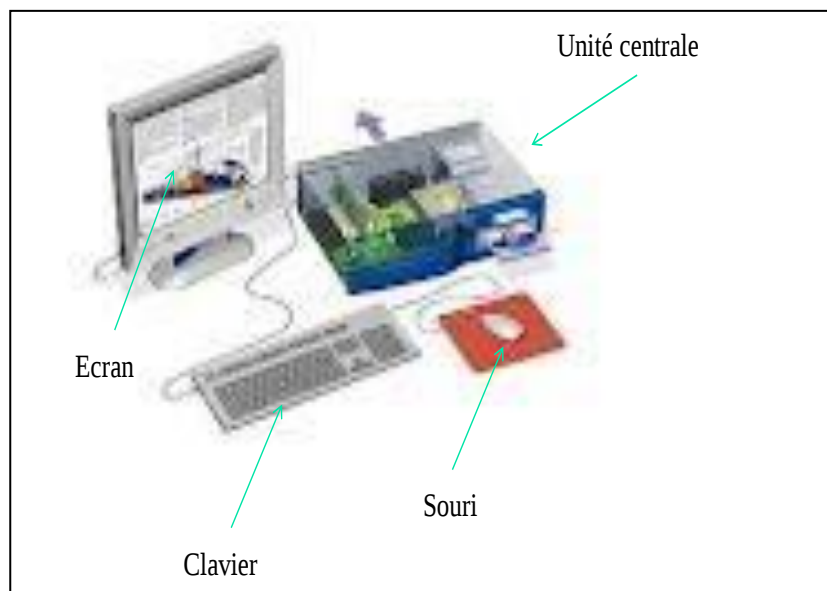


Figure 2: Schéma simplifié d'un ordinateur

https://encryptedtbn0.gstatic.com/images?q=tbn:ANd9GcTiMTWlqkZ34K9rxj2HPA59qHW_CaiU1a8Vlg&usqp=CAU

2. Notions sur le Fonctionnement d'un PC:

- le fonctionnement d'un ordinateur est basé sur un **système informatique**.
- Un **système informatique** a besoin de **matériels** et de **logiciels**.
- **matériels** (les éléments physique de l'ordinateur),
- **logiciels** (ensemble de programme).

a/ Matériels (hardware):

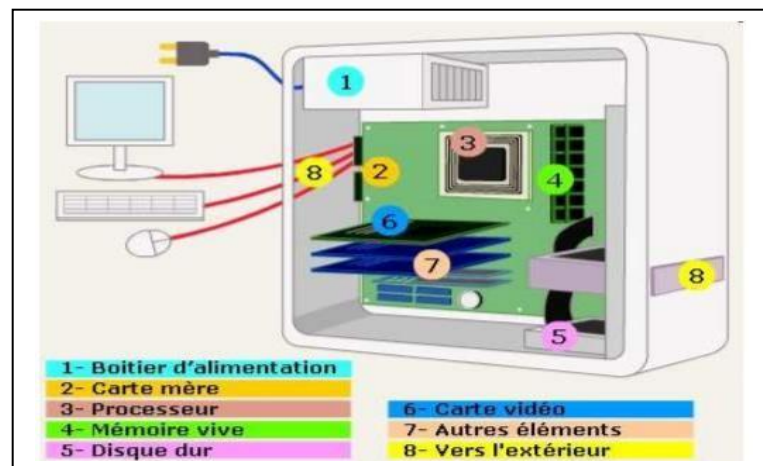


Figure 3: Schéma général de l'unité central

Diaporama_initiation_informatique, L'informatique pour débutants

- La partie matériel est la partie physique et palpable du système informatique.



Figure 4: schéma général de l'ordinateur

Diaporama_initiation_informatique, L'informatique pour débutants

Elle est divisée en deux parties :

-L'unité centrale - Les périphériques

-L'unité centrale : C'est là où s'effectue l'essentiel du traitement de l'information.

Elle comporte:

Le processeur : c'est l'unité d'exécution. permet à l'ordinateur d'effectuer les opérations demandés.



Figure 5: le processeur

<https://apcpedagogie.com/wp-content/uploads/2018/02/le-processeur-300x160.jpg>

La mémoire : c'est l'unité de stockage, elle sauvegarde les informations, On distingue :

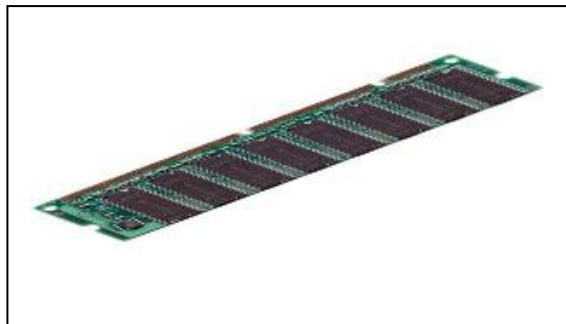


Figure 6: la mémoire

<http://img.overblog.com/300x300/4/24/46/52/SDRAM.jpg>

La mémoire interne : RAM C'est la mémoire principale de l'ordinateur, elle permet de stocker les informations des programmes et les données en cours de fonctionnement. Ce stockage est temporaire, contrairement à une mémoire de masse comme le **disque dur**.



Figure 7: le disque dur

La mémoire externe : c'est celle qui est déplaçable. Exemple : les disquettes, le CD ROM, le DVD ROM, la clé USB,

La carte mère :

- Relie tous les éléments constituant un ordinateur:
Microprocesseur, les barrettes de mémoire et les autres éléments matériels tels que les périphériques d'entrée et de sortie.
- permet de faire circuler les informations entre tous les composants de l'ordinateur.

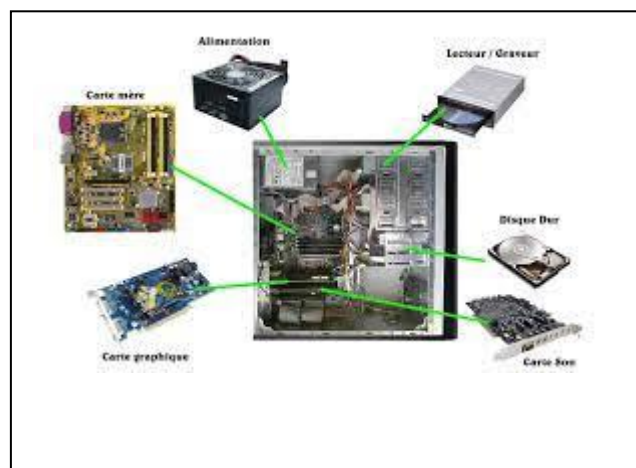


Figure 8: La carte mère

https://encryptedtbn0.gstatic.com/images?q=tbn:ANd9GcT4TwRdQvIO2NJ1Xwt6immw1Zf9gZO63NrifIbZR1hyVuqzxESC6t_9OPABkMc4ErMjTuY&usqp=CAU

Carte graphique: qui permet de produire une image pouvant être affichée sur un écran.

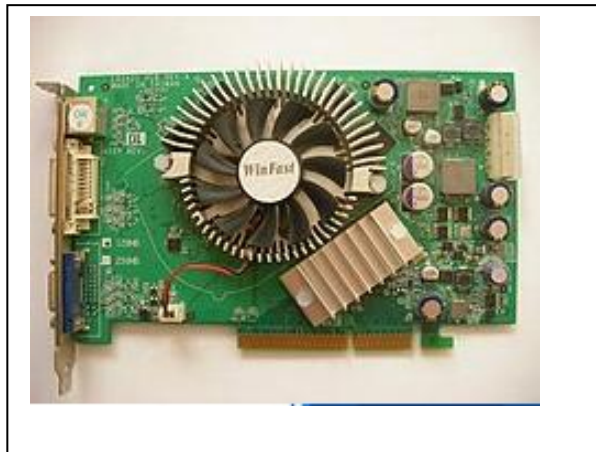


Figure 9: Carte graphique

https://upload.wikimedia.org/wikipedia/commons/thumb/4/42/Carte_graphique_%28Leadtek_WinFast_6600_GT%29.jpg/236pxCarte_graphique_%28Leadtek_WinFast_6600_GT%29.jp

-Les périphériques : on distingue trois classes de périphériques :

Les périphériques d'entrée: saisie des données

Exemple:

- Clavier
- Souris
- Caméra
- Micro
- lecteur de CD-ROM
- Etc.....

Les périphériques de sortie: restituer les données

Exemple:

- Écran
- Imprimante
- Haut-parleur
- Etc...

Les périphériques d'entrées de sortie: saisir et restituer des données

Exemple:

- Graveur DVD
- lecteur de disquette
- Etc.....

b/ Logiciel (software):

- Les logiciels sont les programmes ,ou les instructions , qui indiquent au matériel ce qu'il faut faire.
 - Un logiciel est un ensemble de programmes destiné à un ordinateur ou à un système informatique pour effectuer une tâche ou une fonction particulière.
 - Il y a 2 types de logiciels: les **systèmes d'exploitation** et les **logiciels d'application**.
 - Un **système d'exploitation** est un logiciel destiné à faciliter et à simplifier l'utilisation d'un ordinateur. Il assure l'interface entre le matériel et l'utilisateur. Le système d'exploitation contrôle le fonctionnement des différents éléments matériels.
- Exemple : **Windows xp, Windows 7, Linux, Unix,**
- Un **logiciel d'application** donne à l'ordinateur des instructions pour qu'il réalise des tâches spécifiques (traitement de texte, jeu, calculer la moyenne de la classe, faire un diaporama électronique,.....).
- Exemple : **Word, les logiciel de jeux, Excel, Power Point, les antivirus,**

3. Les réseaux et sécurité informatiques:

-Un **réseau** est un ensemble d'équipements reliés entre eux pour échanger des informations.

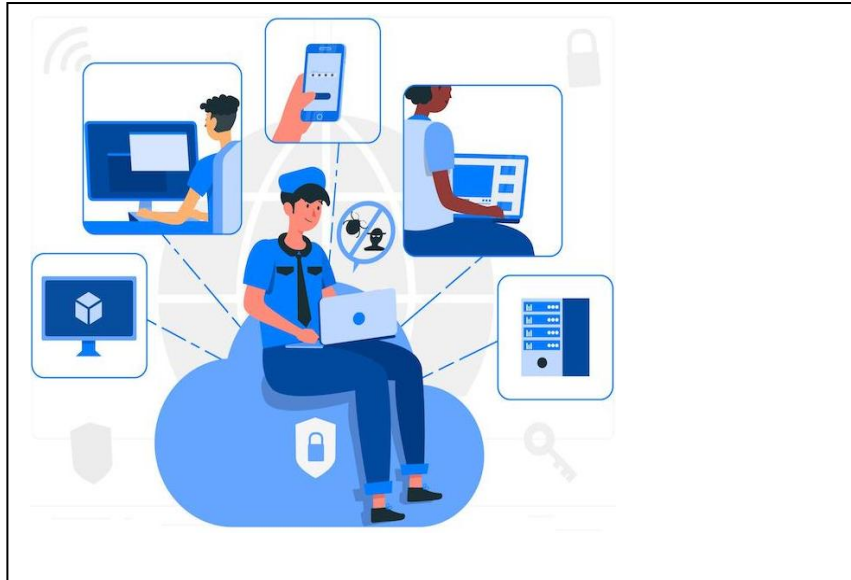


Figure 10: Les réseaux et sécurité informatiques

https://encryptedtbn0.gstatic.com/images?q=tbn:ANd9GcTbTC1_xdKuDnm_Vf5UM5Jsz2ZB77Tgsnu-4g&usqp=CAU

-La sécurité informatique protège l'intégrité des technologies de l'information comme les systèmes, les réseaux et les données informatiques contre les attaques, les dommages ou les accès non autorisés. Pour préserver leur compétitivité dans le contexte de la transformation numérique, les entreprises doivent comprendre comment adopter des solutions de sécurité informatique qui sont intégrées dès la phase de conception. <https://www.redhat.com/fr/topics/security>

- La **sécurité** du réseau est utilisée pour empêcher les utilisateurs malveillants ou non autorisés d'accéder à votre réseau. ...

- **Sécurité** Internet. ...
- **Sécurité** de point d'extrémité ...
- **Sécurité** du nuage. ...
- **Sécurité** des applications.

- Une politique de sécurité réseau doit respecter un ensemble de principes génériques :

- **Identification** : Information permettant d'indiquer qui vous prétendez être. Une identification élémentaire est le nom d'utilisateur que l'on saisit dans un système informatique. Une identification plus évoluée peut être le relevé d'empreinte digitale, l'analyse fiscale, rétinienne, etc.
 - **Authentification** : Information permettant de valider l'identité pour vérifier que vous êtes celui que vous prétendez être. Une authentification élémentaire est le mot de passe que vous entrez dans le système informatique. Une authentification forte combine une chose que vous possédez et une chose que vous connaissez, (numéro de carte bancaire et code personnel, par exemple).
 - **Autorisation** : Information permettant de déterminer quelles seront les ressources de l'entreprise auxquelles l'utilisateur identifié et autorisé aura accès ainsi que les actions autorisées sur ces ressources. Cela couvre toutes les ressources de l'entreprise.
 - **Confidentialité** : Ensemble des mécanismes permettant qu'une communication de données reste privée entre un émetteur et un destinataire. La cryptographie ou le chiffrement des données est la seule solution fiable pour assurer la confidentialité des données.
 - **Intégrité** : Ensemble des mécanismes garantissant qu'une information n'a pas été modifiée.
 - **Disponibilité** : Ensemble des mécanismes garantissant que les ressources de l'entreprise sont accessibles, que ces derniers concernent l'architecture réseau, la bande passante, le plan de sauvegarde, etc.
- Nom réputation** : Mécanisme permettant de garantir qu'un message a bien été envoyé par un émetteur et reçu par un destinataire.
- **Traçabilité** : Ensemble des mécanismes permettant de retrouver les opérations réalisées sur les ressources de l'entreprise. Cela suppose que tout événement applicatif soit archivé pour investigation ultérieure.

<https://www.cours-gratuit.com/cours-reseau/cours-sur-les-principes-de-securite-informatique>.

Le système de numération

1. Introduction :

- **Coder** : c'est écrire dans un code.
- **Le codage de l'information** = passer d'une représentation de l'information à une autre.
- L'information élémentaire est appelée bit.
- Un bit c'est soit 1 soit 0.
- On regroupe les bits par mots de différentes largeurs : mots de 8, 16, 32 ou 64 bits.
- 1 octet = 1 Byte = 8 bits.
- 2. Les différents systèmes de numération (Binaire, Octal, Décimal et Hexadécimal):
- En informatique, les systèmes de numération les plus courants sont :

Système	Base	Symbole	Nombre de symbole
binaire	2	0,1	2
Octal	8	0,1,2,3,4,5,6,7	8
décimal	10	0, 1,2,3,4,5,6,7,8,9	10
hexadécimal	16	0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F	16

Tableau 1: Les différents systèmes de numération

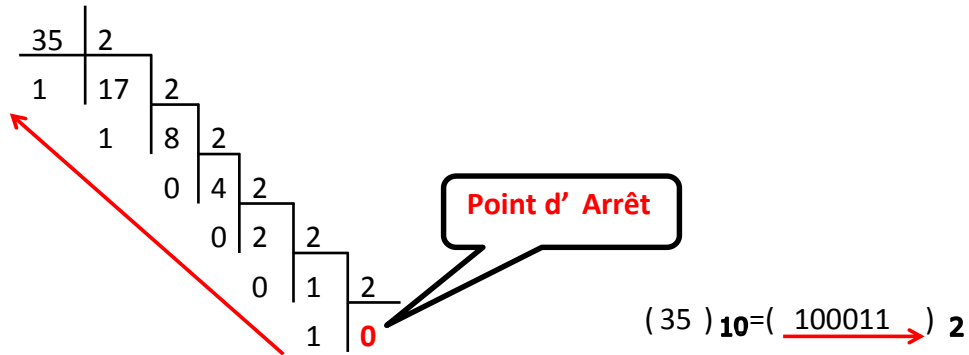
<https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcSTwPZjfZVsLF9Ip-IL8y7BEH4QKlD1UyBIOg&usqp=CAU>

2. Le transcodage : les différentes conversions

➤ Conversion décimale binaire :

Soit le nombre décimal

$$(35)_{10} = (\dots ? \dots)_2$$

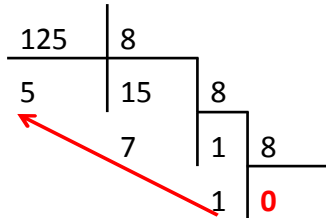


Donc: 35 (décimale) = 100011 (binaire)

➤ Conversion décimale Octal :

Soit le nombre décimal 125:

$$(125)_{10} = (\dots ? \dots)_8$$



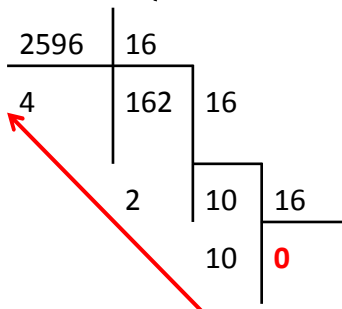
$$(125)_{10} = (175)_8$$

Donc: 125 (décimale) = 175 (Octal)

➤ Conversion décimale hexadécimal :

Soit le nombre décimal 2596:

$$(2596)_{10} = (\dots ? \dots)_{16}$$



$$(2596)_{10} = (1024)_{16}$$

Donc: 2596 (décimale)=1024 (hexadécimale)

➤ **Conversion binaire décimale :**

Soit le nombre binaire : 1101101 sa conversion en décimal est immédiate :

$$(1101101)_2 = 1*2^6 + 1*2^5 + 0*2^4 + 1*2^3 + 1*2^2 + 0*2^1 + 1*2^0$$

$$= 64 + 32 + 8 + 4 + 1 = 109 \text{ (décimal)}$$

Donc $(1101101)_2 = (109)_{10}$

➤ **Conversion Octal décimale:**

Soit le nombre Octal : 175

Sa conversion en décimal est immédiate :

$$(175)_8 = 1*8^2 + 7*8^1 + 5*8^0$$

$$= 64 + 56 + 8$$

$$= 125 \text{ (décimal)}$$

Donc $(175)_8 = (125)_{10}$

➤ **Conversion hexadécimale décimale:**

Soit le nombre hexadécimale : A24 sa conversion en décimal est immédiate :

$$(A24)_{16} = 10*16^2 + 2*16^1 + 4*16^0$$

$$= 2560 + 32 + 4$$

$$= 2596 \text{ (décimal)}$$

Donc $(A24)_{16} = (2596)_{10}$

3. Le transcodage d'un texte en langage binaire:

- Chaque caractère possède donc son équivalent en code numérique: c'est le code ASCII (American Standard Code for Information Inter change).
- Dans le code ASCII Les caractères sont représentés sur 8 bits.

Symbol	Decimal	Binary
A	65	01000001
B	66	01000010
C	67	01000011
D	68	01000100
E	69	01000101
F	70	01000110
G	71	01000111
H	72	01001000
I	73	01001001
J	74	01001010
K	75	01001011
L	76	01001100
M	77	01001101
N	78	01001110
O	79	01001111
P	80	01010000
Q	81	01010001
R	82	01010010
S	83	01010011
T	84	01010100
U	85	01010101
V	86	01010110
W	87	01010111
X	88	01011000
Y	89	01011001
Z	90	01011010

Symbol	Decimal	Binary
a	97	01100001
b	98	01100010
c	99	01100011
d	100	01100100
e	101	01100101
f	102	01100110
g	103	01100111
h	104	01101000
i	105	01101001
j	106	01101010
k	107	01101011
l	108	01101100
m	109	01101101
n	110	01101110
o	111	01101111
p	112	01110000
q	113	01110001
r	114	01110010
s	115	01110011
t	116	01110100
u	117	01110101
v	118	01110110
w	119	01110111
x	120	01111000
y	121	01111001
z	122	01111010

Tableau 2: le code ASCII

https://slideplayer.com/12939748/78/images/slide_1.jpg²²

Travaux Dirigés

Exercice 1 :

Parmi les périphériques suivants, identifiez les périphériques d'entrées, de sorties et d'entrées/sorties :

Lecteur de CD-ROM, haut-parleur, clavier, écran, écran tactile, lecteur DVD, scanner, webcam, graveur DVD, souris, microphone, casque, connecteurs USB.

périphériques d'entrées	périphériques sorties	périphériques d'entrées/sorties

Exercice 2 : Donnez la classe de chacun des logiciels suivants :

Windows XP

Les pilotes de périphériques

Winrar

Mac OS

Excel

Linux

Word

Unix

Les logiciels de jeux

L'antivirus.....

Power point

Microsoft office word

Exercice 3 :

Combien de valeurs peuvent prendre un bit ?

Quelles sont ces valeurs ?

Combien de combinaisons (code binaire) on peut représenter avec 36 bits ?.....

Exercice 4 : convertir les nombres suivants en décimal:

$(10111010)_2 = (\dots\dots\dots)_{10}$

$(45)_8 = (\dots\dots\dots)_{10}$

$(A62)_{16} = (\dots\dots\dots)_{10}$

Exercice 5 : convertir les nombres suivants en binaire :

$(45)_{10} = (\dots\dots\dots)_2$

$(45)_{10} = (\dots\dots\dots)_8$

$(45)_{10} = (\dots\dots\dots)_{16}$

Exercice 6:

Combien de valeurs peuvent prendre un bit ?

Quelles sont ces valeurs ?

Combien de combinaisons (code binaire) on peut représenter avec 36 bits ?.....

On veut donner des codes aux 243 étudiants, déterminez la taille des codes (en nombre de positions) si on considère un codage basé sur:

7 symboles (0, 1, 4, 5, 6, 7,8):.....

4 symboles (a, c, d, g):.....

Exercice 7 : Réaliser les passages suivants :

136 bit =..... Octet

4 TéraOctet =KiloOctet

3072 MégaOctet =.....GigaOctet

27 GigaOctet =..... MégaOctet=..... Octet

245 MégaOctet=..... Octet

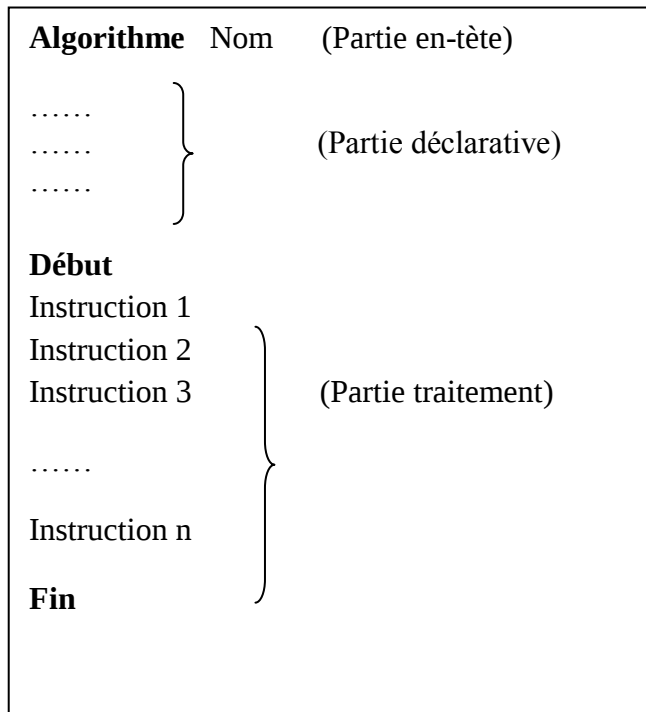
1024 MégaOctet=..... Octet=.....bit

Base de l'Algorithmique

1. Définition du mot Algorithme :

Un **algorithme** est le résultat d'analyse d'un problème, en utilisant des méthodes et des règles mathématiques, pour que celui-ci soit présenté à une machine (ordinateur).

2. La structure d'un algorithme :



- Partie en tête : est définie par le mot clé **algorithme**, le nom de l'algorithme est alphanumérique.
- Partie déclarative : contient la déclaration des types de données.
- Partie traitement (corps de l'algorithme): commence par le mot clé **début** et se termine par le mot clé **fin**.

3. Constante: c'est une case mémoire invariable durant le déroulement d'un algorithme.

-Déclaration d'une constante :

Constante nom Constante = valeur

-Exemple : **Constante** PI = 3,14

Constante deux= '2'

La constante 2 est considéré comme un caractère, on l'additionne pas avec les nombres, on la concatène avec les caractères.

4. **Variable**: c'est une case mémoire qui peut être modifié durant le déroulement d'un algorithme.

-Déclaration d'une variable :

Variable nom Variable : type
--

-Exemple : **Variable** note : réel -> note 13.5

Variable coef : entier -> coef 5

Variable A : booléen -> A faux

Variable Nom: chaine -> Nom Mohammed

Variable B : caractère -> B c

-Remarque : dans un algorithme, on déclare toujours les constantes (*s'il ya des constantes dans l'algorithme*) avant les variables.

5. **Les types de variable** :

- Entier: l'ensemble des **entiers** -> **Variable** coef : entier

- Réel : l'ensemble des **réels** -> **Variable** note : réel

- Booléen : les valeurs **vrai** et **faux** -> **Variable** B : Booléen

- Caractère -> **Variable** C : caractère

Constante A= '3'

- chaine de caractère: ensemble de caractères, alphanumérique.

6. **Les instructions** :

A/ **Instruction d'affectation** :

Une affectation est une instruction qui stocke dans une variable une valeur.

Variable ← valeur constante ou Expression arithmétique ou Expression logique.

Exemple :

- **valeur constante :**

$A \leftarrow 15$: la variable A de type entier reçoit la valeur 15.

A

15

$B \leftarrow 2$: la variable B de type entier reçoit la valeur 2.

B

2

- **Expression arithmétique :**

$S \leftarrow A+B$: la variable S de type entier reçoit l'expression arithmétique A+B, donc S contient la valeur 17.

C

17

- **Expression logique :**

$C \leftarrow \text{vrai}$: la variable C de type booléen reçoit vrai.

B/ **Instruction de lecture** : permet d'introduire des valeurs aux variables par le clavier.

Lire (variables)

Exemple : Lire (K) -> donner une valeur à K.

Lire (a,b,c) -> donner une valeur à a et b et c

C/ **Instruction d'écriture** : permet d'afficher un résultat sur l'écran.

Écrire (variables)

-exemple :

$A \leftarrow 15$

$B \leftarrow 2$

Écrire (A)

Écrire ('Bonjour')

Écrire (B)

Écrire ('B')

Écrire ('donner une valeur à A et B, ')

-Solution :

15
Bonjour
2
B
donner une valeur à A et B

Ecran

Travaux Dirigés

Exercice 1 :

Dérouler l'Algorithme suivant et donner le résultat sur écran :

```
Algorithme déroulement
Variable x,y,z : entier
           a,b : chaîne de caractères

Début
Lire (x)
x←16
y←x+11
lire (y)
y←5
z←x-y
a←'salut'
b←a
ecrire (x,y,z)
ecrire(b)
ecrire('b')
Fin.
```

Exercice 2 :

Donner l'algorithme qui calcule la somme de 2 nombres entiers.

Exercice 3 :

Donner l'algorithme qui calcule la moyenne de 4 variables x,y,z,w.

Exercice 4 :

Donner l'algorithme qui calcule la moyenne générale d'un étudiant. Cette moyenne générale est calculée en utilisant 4 notes de 4 modules différents. Les coefficients des 4 modules sont respectivement coef1, coef2, coef3, coef4.

Exercice 5 :

Donner l'algorithme qui calcule le prix de vente d'une marchandise se trouvant dans un magasin, connaissant son prix d'achat, le taux de la valeur ajoutée et le taux du bénéfice.

D/Les structures de contrôles :

- Introduction :

Un traitement peut s'exécuter de différentes manières :
- séquentiellement
- alternativement
- répétitivement

- Le traitement séquentiel :

C'est une liste d'instructions qui s'exécutent l'une à la suite des l'autres (vue dans le TD1).

- Le traitement alternatif : (Les instructions conditionnelles)

1/ **Alternatif simple :** (*Structure conditionnelle simple*)

..... Si condition alors Liste d'instructions Finsi
--

-**Principe** : si la condition (Expression booléenne) est évaluée à VRAI, alors la liste d'instruction s'exécute. Sinon Si la condition est évaluée à FAUX on exécute directement la suite de l'algorithme (allez directement à **Finsi**).

-Exemple :

Si $A > B$ **alors**

Ecrire ('A est supérieure à B')

Fsi

1/ Donner le résultat pour $A=5$ et $B=8$,

2/ pour $A=10$ et $B=9$.

-Solution :

1/ Pour $A=5$ et $B=8$	2/ Pour $A=10$ et $B=9$
------------------------	-------------------------

2/ Alternatif composé : (Structure conditionnelle alternative)

```
.....  
Si condition alors  
        Liste d'instructions 1  
  
Sinon  
        Liste d'instructions 2  
  
Finsi  
.....
```

-Principe : si la condition (Expression booléenne) est évaluée à VRAI, alors la liste d'instruction 1 s'exécute. Sinon Si la condition est évaluée à FAUX alors la liste d'instruction 2 s'exécute, puis on exécute la suite de l'algorithme.

-Exemple :

Si $A > B$ **alors**

Ecrire ('A est supérieure à B')

Sinon

Ecrire ('B est supérieure à A')

Finsi

1/ Donner le résultat pour $A=5$ et $B=8$,

2/ pour $A=10$ et $B=9$.

-Solution :

1/ Pour $A=5$ et $B=8$

2/ Pour $A=10$ et $B=9$

3/ Imbrications de conditions :

```
.....  
Si condition1 alors  
    Liste d'instructions 1  
Sinon  
    Si condition 2 alors  
        Liste d'instructions 2  
    Sinon  
        Si condition 3 alors  
            Liste d'instructions 3  
        Sinon  
            Liste d'instructions 4  
        Finsi  
    Finsi  
Finsi  
.....
```

Travaux Dirigés

Exercice1 :

Algorithme Alternatif simple

Variable a,b,c,d,e : entiers

Debut

Lire (a,b)

$c \leftarrow 0$

$d \leftarrow 0$

si $a > b$ alors

$c \leftarrow a + b$

$d \leftarrow a - b$

Finsi

$e \leftarrow a * b$

Ecrire (c,d,e)

Fin

- Dérouler l'algorithme pour :

1/ $a=2$ et $b=4$

2/ $a=5$ et $b=4$

Exercice 2 :

Ecrire l'algorithme qui calcul R selon les cas suivants :

-Si $x > y$: $R = a * b$

-Si $x = y$: $R = a + b$

Exercice 3 :

Ecrire un algorithme qui permet de saisir deux nombres entiers x,y et les afficher à l'écran dans l'ordre croissant.

Exercice 4 :

Ecrire un algorithme qui affiche l'état de l'eau selon les différentes températures :

-C'est de la glace : Température ≤ 0

-C'est du liquide: Température < 100

-C'est de la vapeur : Température ≥ 100

E/ Le traitement répétitif (itératif): (les boucles)

Le traitement itératif permet de répéter le bloc d'instructions plusieurs fois.

-Remarque :

- **I** : Une variable entière appelée **compteur** sert à compter le nombre de répétitions du bloc d'instructions.
 - Chaque boucle à une valeur initiale et une valeur finale.
 - **I** ← 1 : initialisation.
 - **I** ← 1+1 : incrémentation.
- } Dans **tant que** et **répéter**.

1/ La boucle pour :

Pour <compteur> allant de <valeur initiale> à <valeur finale> **Faire**

Bloc d'instructions

Fin pour

-Principe : pour compteur (I) allant de valeur initiale à valeur finale exécuté le bloc d'instruction.

-Exemple : écrire un algorithme qui affiche les 100 premiers nombres positifs en utilisant la boucle **pour**.

-Solution :

2/ La boucle tant que :

Tant que <condition vérifiée> **Faire**

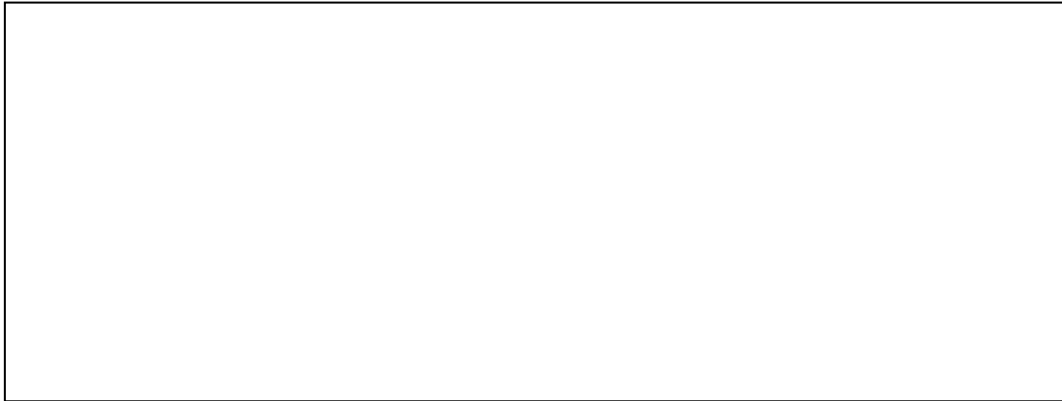
Bloc d'instructions

Fin tant que

-Principe : Tant que la condition est vérifiée (vraie), on répète l'exécution du bloc d'instructions qui est à l'intérieure de la boucle. Dès que la condition est fausse on sort de la boucle et on continue la suite des instructions qui sont après la boucle. Si la condition est initialement fausse le bloc d'instruction ne sera jamais exécuté.

-Exemple : écrire un algorithme qui affiche les 100 premiers nombres positifs en utilisant la boucle **tan que**.

-Solution :



3/ La boucle répéter :

Répéter

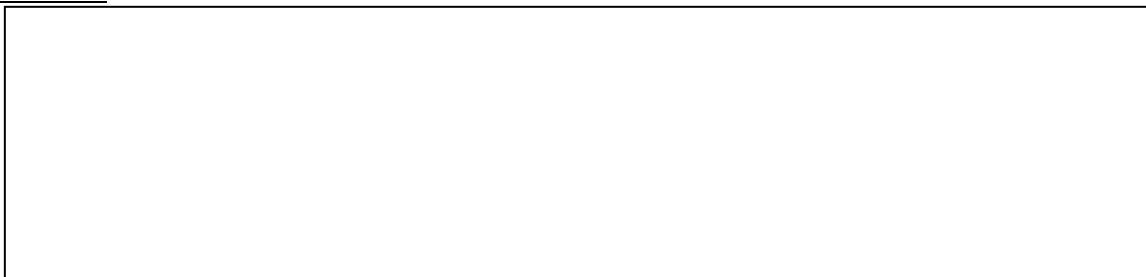
Bloc d'instructions

Jusqu'à <condition vérifiée>

-Principe : Elle permet de répéter un bloc d'instructions jusqu'à ce qu'une condition soit satisfaite. Le bloc d'instructions est exécuté au moins une fois.

-Exemple : écrire un algorithme qui affiche les 100 premiers nombres positifs en utilisant la **répéter**.

-Solution :



Travaux Dirigés

Exercice1 :

Algorithme boucle **pour**

Variable n,p,I :entier

Debut

Lire(n)

$p \leftarrow 1$

Pour I =1 à n faire

$p \leftarrow p * 2 * I$

ecrire (p)

Finpour

Fin.

1/ Dérouler l'algorithme pour n=4.

2/ Ecrire cet algorithme en utilisant la boucle **tant que**, dérouler l'algorithme pour n=4.

3/ Ecrire cet algorithme en utilisant la boucle **répéter**, dérouler l'algorithme pour n=4.

Exercice 2 :

Ecrire un algorithme qui calcule et affiche le produit suivant :

$$P = 1 * 2 * 3 * 4 * 5 * \dots * n.$$

Exercice 3:

Ecrire un algorithme qui calcule et affiche la somme suivante :

$$S = 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5} + \dots + \frac{1}{n}.$$

Exercice 4:

Soit n étudiants, écrire un algorithme qui permet d'introduire 3 notes pour chaque étudiant, et calculer la moyenne tel que leurs coefficients sont comme suit :1,2,3 et l'algorithme donne les résultats suivantes :

-Si $\text{moy} \geq 10$: « admis ».

-Si $\text{moy} \leq 8$: « ajourné ».

-Si $8 < \text{moy} < 10$: « repassé ».

Présentation Turbo-Pascal Windows

1. Un programme :

- Un programme est un ensemble d'instructions (opérations) destinées à être exécutées par un ordinateur.
- Un programme source est un code écrit par un informaticien dans un **langage de programmation** (exemple Pascal) . Il peut être **compilé** vers une forme binaire ou directement interprété.

2. Structure générale d'un programme PASCAL :

Program nom du programme ;

Uses Wincrt ;

Const identificateur= valeur ;

Var Identificateur : type ;

Begin

Instruction 1 ;

.

.

.

Instruction n ;

End.

3. La déclaration des constantes:

Const identificateur =valeur ;

Exemple :

Const g=10 ;

Prénom ='Mohamed' ;

4. La déclaration des variables:

Var identificateur1, identificateur2 : type1 ;

Exemple :

Var S,M : real;

I,J : integer ;

5. Les types de donnée de base dans Pascal :

Integer : entier

Real : réel

Char : caractère alphanumérique.

String : chaîne de caractères.

Boolean : valeurs logiques (TRUE ou FALSE).

-Remarque sur l'**identificateur**: l'identificateur doit être :

Un alphanumérique qui commence par un alphabétique.

Ne doit pas contenir de ponctuation ou d'accent.

Le Pascal ne fait pas la différence entre les majuscules et les minuscules

6. Instructions élémentaires :

• Affectation :

L'affectation en Pascal est comme suit :=

Variable := valeur ;

-Remarque :

La variable et la valeur doivent avoir le même type.

On peut avoir une valeur de type entier affectée à une variable de type réel.

-Exemple :

a1:= 9 la variable a1 prend la valeur 9

a2 := 4 la variable a2 prend la valeur 4

a3 := a1+a2 la variable a3 prend le résultat de la somme de a1et a2 c à d (13)

x2:= x1 la variable x2 prend la valeur de x1

c := 'h' la variable c prend le caractère h

• Les instructions READ/WRITE :

✓ L'instruction write() :

L'instruction **write** permet d'afficher des informations à l'écran. On peut s'en servir pour écrire :

- une chaine de caractère : **write**('Donner la valeur de A')
- la valeur d'une variable(le contenu d'une variable) : **write**(x)

- une chaîne de caractère+ la valeur d'une variable(le contenu d'une variable): **write('x=',x)**

writeln() passe pas à la ligne suivante à la fin de la commande.

L'instruction **read()** :

L'instruction **read(x1, . . . , xn)** ; permet de transmettre des données au programme,entrées au clavier. Ces données doivent être séparées par un caractère espace ou un retour à la ligne.

readln() passe pas à la ligne suivante à la fin de la commande.

-Expressions sur les objets :

Une expression est une suite d'opérandes reliés par des opérateurs. Une expression peut être :

- ✓ **Arithmétique** : on utilisera les opérateurs +,-,/,*,div, mod.
- ✓ **Logique** : les opérateurs utilisés sont and, or et not
- ✓ **Relationnelle** : les opérateurs utilisés sont <, <=, >, >=, =, <>

- **Les instructions conditionnelles:**

Cette instruction permet d'exécuter le bloc d'instruction1 si et seulement si la condition (Expression booléenne) est évaluée à VRAI. La syntaxe de cette instruction en langage Pascal est la suivante :

Une seule instruction	Plus d'une instruction
IF condition then Instruction 1 ;	IF condition then begin Instruction 1 ; Instruction 2 ; . . End ;

- **Structure conditionnelle avec alternative:**

Cette instruction permet d'exécuter le bloc d'instruction1 si et seulement si la condition (Expression booléenne) est évaluée à VRAI. Si la condition est évaluée à FAUX on exécute le bloc d'instruction2. La syntaxe de cette instruction en langage Pascal est la suivante :

Une seule instruction	Plus d'une instruction
IF condition then Instruction 1 else Instruction 2 ;	IF condition then begin Instruction 1 ; Instruction 2 ; . . End Else begin Instruction 1 ; Instruction 2 ; . . End ;

Remarque : avant **else**, il n'y a pas de point virgule « ; »

- **Imbrications de conditions :**

nous pouvons imbriquer plusieurs conditions les unes à la suite des autres. La syntaxe de cette instruction en langage Pascal est la suivante :

Une seule instruction	Plus d'une instruction
IF condition1 then Instruction 1 else IF condition 2 then Instruction 2 . . . else Instruction n ;	IF condition 1 then begin Instruction 1 ; Instruction 2 ; . . End Else IF condition 2 then begin Instruction 1 ; Instruction 2 ; . . End Else begin Instruction 1 ; Instruction 2 ; . . End ;

- **Les boucles :**

- **For.....do :**

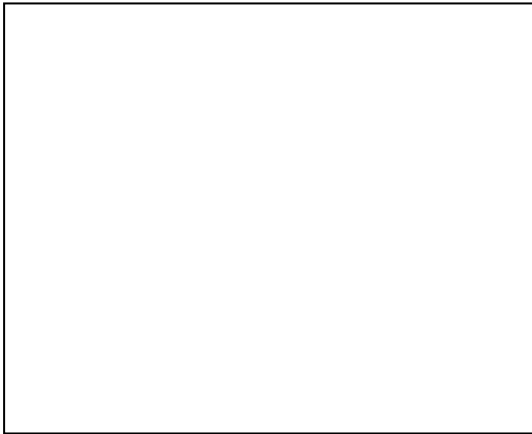
La syntaxe de la boucle for est données par :

Une seule instruction	Plus d'une instruction
For I :=1 to n do Instruction 1 ;	For I :=1 to n do begin Instruction 1 ; Instruction 2 ; . . End ;

Exercice1 :

En utilisant la boucle **For**, écrire un programme qui affiche les 100 premiers nombres entiers positifs.

Solution1 :



○ **Whiledo :**

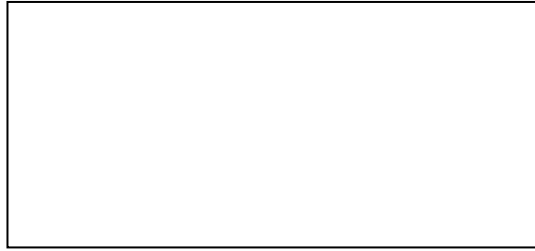
Les instructions sont exécutées si la condition= vrai, la syntaxe de la boucle while est données par :

Une seule instruction	Plus d'une instruction
While condition do Instruction 1 ;	While condition do begin Instruction 1 ; Instruction 2 ; . . End ;

Exercice2 :

En utilisant la boucle **While**, écrire un programme qui affiche les 100 premiers nombres entiers positifs.

Solution 2 :



○ **Repeat...until :**

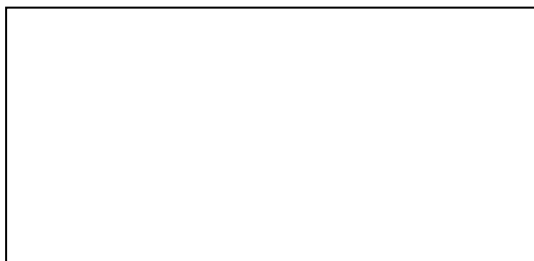
Les instructions sont exécutées si la condition= faux, la syntaxe de la boucle repeat est donnée par :

Une seule instruction	Plus d'une instruction
Repeat Instruction 1 ; Until condition ;	Repeat Instruction 1 ; Instruction 2 ; . . Until condition ;

Exercice 3:

En utilisant la boucle **Repeat**, écrire un programme qui affiche les 100 premiers nombres entiers positifs.

Solution 3 :



Travaux Dirigés

Exercice n°01 :

Ecrire en pascal un programme qui calcule la valeur absolue de X

Exercice n°02 :

Écrire un programme en pascal qui calcule le carré et le double d'une valeur saisie au clavier.

Exercice n°03:

Ecrire un programme en pascal qui demande un nombre à l'utilisateur, et l'informe ensuite si ce nombre est positif ou négatif (on laisse de côté le cas où le nombre vaut zéro).

Exercice 4 :

Ecrire un programme en pascal qui demande un nombre de départ, et qui calcule sa factorielle.

NB : la factorielle de 8, notée 8 !, vaut $1 \times 2 \times 3 \times 4 \times 5 \times 6 \times 7 \times 8$

Les structures de Données Complexes

1. Les Tableaux à une dimension :

1.1. Définition d'un tableau :

Un **tableau** est une liste ordonnée de n valeurs du même type. On appelle n la taille du **tableau**, et les valeurs qu'ils contiennent sont ses éléments. Chaque élément est repéré dans le **tableau** par son **indice**.

1.2. La déclaration d'un tableau à une dimension :

Un tableau se déclare comme suit :

var nom du tableau : **tableau** [indice minimal..indice maximal] **de** type

exemple : **var T : tableau [1..5] d'entier**

1.3. Manipulation d'un tableau :

-Un tableau peut être représenté par un ensemble de cases, l'une à côté de l'autre, schématisé comme suit :

Indice	1	2	3	4	5
Contenu					

Représentation du tableau **T**

-Un élément du tableau (le contenu d'une case) est accessible par sa position (indice) dans le tableau. Le tableau **T** est un tableau à une seule dimension là où chaque élément est accessible par un seul indice. L'accès peut être pour une lecture, écriture, modification, etc.

Par exemple, pour affecter une valeur au premier élément du tableau, on met :

T[1] :=5 ; et le tableau **T** devient :

Indice	1	2	3	4	5
Contenu	5				

Remarque :Pour la manipulation des tableaux, on utilise fréquemment les boucles.

1.4. La lecture d'un tableau à une dimension :

pour I :=1 à n faire lire (T[I])

1.5. L'affichage d'un tableau à une dimension :

Pour I :=1 à n faire write(T[I])

Travaux dirigés

Tableau à une dimension

Exercice1 :

Ecrire un algorithme qui permet de remplir un tableau T(100) dont les éléments sont des entiers, puis afficher le contenu de ce tableau.

Exercice 2 :

Ecrire un algorithme qui permet de faire la somme des éléments d'un tableau d'entiers.

Exercice 3 :

Soient T1 et T2 deux tableaux de même longueur, construire un tableau $T3=T1+T2$.

Exercice 4 :

Ecrire un algorithme qui permet de rechercher le plus petit élément dans un tableau a une dimension A(50).

Exercice 5 :

Ecrire un algorithme qui permet de rechercher le plus grand élément dans un tableau a une dimension A(25).

Exercice 5 :

Soit A un tableau d'entier, écrire un algorithme qui permet de compter le nombre d'apparition d'une valeur VAL donnée en donnant sa position.

Exercice 6 :

Soit A un tableau de caractères, comptez le nombre de répétition du caractère 'e'.

Exercice 7 :

Ecrire un algorithme qui détermine le nombre d'éléments (>0) et (<0) et (=0), dans T(50).

Exercice 8 :

Ecrire un algorithme qui détermine le nombre d'éléments (>0) et (<0) et (=0), dans T(50).

Exercice 9:

Soit T un tableau d'entiers contient 100 éléments, écrire un algorithme qui permet de trier le tableau par ordre croissant.

2. Les tableaux à deux dimensions :

Un tableau à deux dimensions est un tableau ayant plusieurs lignes et plusieurs colonnes.

7 colonnes

5 lignes

A[1,1]	A[1,2]	A[1,3]	A[1,4]	A[1,5]	A[1,6]	A[1,7]
A[2,1]	A[2,2]	A[2,3]	A[2,4]	A[2,5]	A[2,6]	A[2,7]
A[3,1]	A[3,2]	A[3,3]	A[3,4]	A[3,5]	A[3,6]	A[3,7]
A[4,1]	A[4,2]	A[4,3]	A[4,4]	A[4,5]	A[4,6]	A[4,7]
A[5,1]	A[5,2]	A[5,3]	A[5,4]	A[5,5]	A[5,6]	A[5,7]

$A[I,j]$

2.1. La déclaration d'un tableau à deux dimensions :

Var nom de la matrice : **tableau**[1..n,1..m] de type

Exemple : **var** M :**tableau**[1..20,1..50] de réel

2.2. La lecture d'un tableau à deux dimensions :

pour I :=1 à n faire

 pour j :=1 à m faire

 lire M[I,j]

 finpour

finpour

2.3. L'affichage d'un tableau à deux dimensions :

pour I :=1 à n faire

 pour j :=1 à m faire

 ecrire M[I,j]

 finpour

finpour

Travaux dirigés

Tableau à deux dimensions

Exercice1 :

Ecrire un algorithme qui permet de remplir un tableau à deux dimensions contenant 30X20 nombres d'entiers, puis afficher le contenu de ce tableau.

Exercice 2 :

Ecrire un algorithme qui permet de faire la moyenne des éléments d'une matrice $M(30,50)$ dont les éléments sont réels.

Exercice 3 :

Ecrire l'algorithme qui permet de faire la somme des éléments de la première diagonale principale d'une matrice carrée.

Exercice 4 :

Ecrire le algorithme qui permet de déterminer l'indice de ligne et l'indice de colonne du plus petit élément d'une matrice $B(n,m)$, dont les éléments sont des entiers.

Exercice 5 :

Soit une matrice $A(n,m)$ d'entiers, écrire un algorithme qui permet de :

1. Compter le nombre de valeurs positives de chaque colonne et calculer leurs sommes.
2. comptez le nombre de valeurs négatives de chaque ligne et calculer leurs produits.

Bloc : Culture générale

Année universitaire :22/23

Niveau : Première

Année (S1)

Unité : Informatique

Durée : 1heure

Examen n°1

Nom :.....

Prénom (s) :.....

Numéro de la carte :.....

Groupe :.....

Exercice 1 : (p) Déterminer les erreurs qui se trouvent dans ce programme :

Program 2	erreur ;
Var i :entier ;	
C : caractère ;	
X,i :réel	
began	
Write(entrez un nombre entier) ;	
X := 5	
c := i ;	
Red('x= ', x)	
And.	

Exercice 2 : (p)

Program exercice2 ;

Uses wincrit ;

Var n,I : integer ;

Begin

Read(n) ;

For I=1 to n do

If (n mod 2)=0 then

Write(I) ;

End.

- 1) Dérouler le programme pour $n = 10$
- 2) Que fait ce programme ?
- 3) Réécrire le programme exercice2 avec le 2 autres boucle

Exercice 3 : (4p) réaliser, les passages suivants (détailler les calculs):

1. $(10111)_2 = (\quad)_{10}$

2. $()_{10} = (\quad)_2$

3. $()_8 = (\quad)_{10}$

4. $()_{10} = (\quad)_8$

Bloc : Culture générale Année universitaire :22/23 Niveau : Première Année (S1)

Unité : Informatique Durée : 1heure

Examen n°1

Nom :.....

Prénom (s) :.....

Numéro de la carte :.....

Groupe :.....

Exercice 1 : (6 p) Déterminer les erreurs qui se trouvent dans ce programme :

Program <u>2erreur</u> ;	Nom du Pg n'est pas alphanumérique
	Uses wincrt
Var i : <u>entier</u> ;	integer
C : <u>caractère</u> ;	char
X,i : <u>réel</u>	real ;
<u>began</u>	begin
Write(entrez un nombre entier) ;	('...')
X := 5	;
c := i ;	
Read('x= ', x)	;
And.	end

Exercice 2 : (8p)

Program exercice2 ;

Uses wincrit ;

Var n,P,I : integer ;

```

Begin
Read(n) ;
P :=1 ;
For I=1 to n do
  begin
    P :=P*2*I ;
    Write(P) ;
  End ;
End.

```

1) Dérouler le programme pour n = 7

n	p	I
7	1	
7	2	1
7	8	2
7	48	3
7	384	4
7	3840	5
7	46080	6
7	645120	7

2, 8, 48, 384, 3840, 46080, 645120

Ecran

2) Que fait ce programme ?

Calcule et affiche le produit de nombres pairs.

3) Réécrire le programme exercice2 avec les 2 autres boucles

While	repeat
Program exercice2 ; Uses wincrit ;	Program exercice2 ; Uses wincrit ;

<pre> Var n,P,I : integer ; Begin Read(n) ; P :=1 ; I :=1 ; While I<= n do begin P :=P*2*I ; Write(P) ; I :=I+1 ; End ; End.</pre>	<pre> Var n,P,I : integer ; Begin Read(n) ; P :=1 ; I :=1 ; Repeat P :=P*2*I ; Write(P) ; I :=I+1 ; Until I> n ; End.</pre>
---	--

Exercice 3 : (6p) réaliser, les passages suivants (détailler les calculs):

1. $(00111101)_2 = (\quad 61 \quad)_{10}$	2. $(7014)_8 = (\quad 3596 \quad)_{10}$
3. $(117)_{10} = (\quad 01110101 \quad)_2$	4. $(137)_{10} = (\quad 211 \quad)_8$

Bloc : Culture générale Année universitaire :21/22 Niveau : Première Année (S1)

Unité : Informatique Durée : 1heure

Examen n°1

Nom :.....

Prénom (s) :.....

Numéro de la carte :.....

Groupe :.....

Exercice 1 : (8p) relier les mots de la colonne A avec la définition de la colonne B :

A	B
-Un système informatique.	• l'unité d'exécution
-La mémoire interne RAM.	• Codage
-La mémoire externe.	• permet de faire circuler les informations entre tous les composants de l'ordinateur
-La carte mère.	• matériels + logiciels
-La mémoire interne ROM.	• Windows 7, Linux, Unix, Word, Excel
-Les logiciels.	• Son contenu est volatile
-Le processeur.	• la clé USB, les disquettes, le CD ROM, le DVD ROM
-traiter l'information.	• mémoire morte

Exercice 2 : (3p) entourer l'intrus dans chaque ligne :

1. Windows, Scanner , Unix, Word, Winrar, Excel.

2. Écran, Imprimante, Caméra, Haut-parleur.

3. La clé USB, les disquettes, le CD ROM, la RAM, le DVD ROM.

Exercice 3: (4p) dérouler l'Algorithme suivant et donner le résultat sur écran :

Algorithme exo 4
Var X, C, D : réel
Début
Ecrire ('donner la valeur à
calculer')
Lire (X)
 $X \leftarrow 12$
 $C \leftarrow X * X$
 $D \leftarrow 2 * X$
Ecrire ('C')
Ecrire (C)
Ecrire ('D')
Ecrire (D)
Fin.

Exercice 4 : (5p) réaliser, les passages suivants (détailler les calculs):

1. $(489)_{10} = (\quad)_2$
2. $(489)_{10} = (\quad)_8$
3. $(489)_{10} = (\quad)_{16}$
4. $(75)_8 = (\quad)_{10}$
5. $(C75)_{16} = (\quad)_{10}$

Nom :.....

Prénom (s) :.....

Numéro de la carte :.....

Groupe :.....

Exercice 1 : (8p)

relier

Un système informatique	• matériels + logiciels
La mémoire interne RAM	• Son contenu est volatile
La mémoire externe	• la clé USB, les disquettes, le CD ROM, le DVD ROM
La carte mère	• permet de faire circuler les informations entre tous les composants de l'ordinateur
La mémoire interne ROM	• mémoire morte
Les logiciels	• Windows 7, Linux, Unix, Word, Excel
Le processeur	• l'unité d'exécution
traiter l'information	• codage

Exercice 2: (3p)

entourer l'intrus dans chaque ligne

1.Windows, Scann , Unix, Word, Winrar, Excel.

2. Écran, Imprimante, Caméra , Haut-parleur.

3. la clé USB, les disquettes, le CD ROM, la RAM , le DVD ROM

Exercice 3: (4) dérouler l’Algorithme suivant et donner le résultat sur écran :

Algorithme exo 4

Var X, C, D : réel

Début

Ecrire ('donner la valeur à calculer')

Lire (X)

$X \leftarrow 12$

$C \leftarrow X * X$

$D \leftarrow 2 * X$

Ecrire ('C')

Ecrire (C)

Ecrire ('D')

Ecrire (D)

donner la valeur à calculer

C (0,5)

144 (0,5)

D (0,5)

288 (0,5)

(0,5)

X	C	D
12	144	288

(0,5) (0,5) (0,5)

Exercice3 : (5) Réaliser, les passages suivants (détailler les calculs):

1. $(489)_{10} = (\quad)_2$ **0,5** + **0,5** pour le détaille

2. $(489)_{10} = (\quad)_8$ **0,5** + **0,5** pour le détaille

3. $(489)_{10} = (\quad)_{16}$ **0,5** + **0,5** pour le détaille

4. $(75)_8 = (\quad)_{10}$ **0,5** + **0,5** pour le détaille

5. $(C75)_{16} = (\quad)_{10}$ **0,5** + **0,5** pour le détaille

Ressources bibliographiques

1. Mr. FEZZA S. 1ère année tronc commun des Sciences Économiques et des Sciences de Gestion et des Sciences Commerciales parcours LMD Semestre 1
URL du site du cours : <https://sites.google.com/site/studentfezza>
2. https://sites.google.com/site/otmani1981/_/rsrc/1470336890657/initiation-a-l-informatique
3. https://encryptedtbn0.gstatic.com/images?q=tbn:ANd9GcTiMTWlqkZ34K9rxj2HPA59qHW_CaiU1a8Vlg&usqp=CAU
4. https://upload.wikimedia.org/wikipedia/commons/thumb/4/42/Carte_graphique_%28Leadtek_WinFast_6600_GT%29.jpg/236px-Carte_graphique_%28Leadtek_WinFast_6600_GT%29.jpg
5. <https://apcpedagogie.com/wp-content/uploads/2018/02/le-processeur-300x160.jpg>
<http://img.overblog.com/300x300/4/24/46/52/SDRAM.jpg>
6. <https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.universitylib.com%2Fcourses-de-systemes-de-numeration->
7. https://www.youtube.com/watch?v=_LtD_r2AWvQ
8. <https://www.youtube.com/watch?v=z9WVh2K0svU>
9. <https://www.gladir.com/SOFTWARE/TP/presentation.htm>
10. <https://www.gladir.com/CODER/TPASCAL7/intro.htm>
11. <https://www.youtube.com/watch?v=UMjqdWxnf5c>
12. https://encryptedtbn0.gstatic.com/images?q=tbn:ANd9GcTbTC1_xdKuDnm_Vf5UM5Jsz2ZB77Tgsnu-4g&usqp=CAU
13. https://slideplayer.com/12939748/78/images/slide_1.jpg²²
14. https://encryptedtbn0.gstatic.com/images?q=tbn:ANd9GcT4TwRdQvIO2NJ1Xwt6immw1Zf9gZO63NrifIbZR1hyVuqzxESC6t_9OPABkMc4ErMjTuY&usqp=CAU