

**Les annales de la programmation
structurée (Algorithme et langage C,
Java, VBNet, C#)**

(Recueil des exercices posés dans les examens de passage à la 2^{ème} année de la filière TDI)

ALGORITHME

C# **C**
Programmation
VBNET
structurée
JAVA
C++

Abdel

HARCHI Abdellah
(Formateur TDI)

Introduction

Abdellah HARCHI

Exercice 1 : Ecrire un algorithme qui demande à l'utilisateur de saisir sa date de naissance (Jour, Mois et Année) ; puis l'algorithme affiche le jour de sa naissance

Remarque : le nombre de jours des mois de l'année (avec simplification) sont indiqués dans le tableau suivant :

Mois	1	2	3	4	5	6	7	8	9	10	11	12
N_Jours	31	28	31	30	31	30	31	31	30	31	30	31

Exercice 2 : Soit un système de grade où $A \geq 18$, $18 > B \geq 16$, $16 > C \geq 14$, $14 > D \geq 12$ et $E < 12$. Ecrivez un programme qui affiche votre grade en fonction de vos points. Les points sont entrés par l'utilisateur et la conversion des points vers les grades doit être faite à l'aide d'une méthode ! Ecrivez le programme en entier.

Exercice 3 : Créez un tableau de 10 nombres entiers dont les valeurs sont entières par l'utilisateur (saisie clavier !) et affichez à l'écran le plus grand et le plus petit nombre ainsi que la moyenne.

Exercice 4 : Tri par insertion

Ecrire un programme permettant de trier suivant l'ordre croissant un tableau à une dimension contenant des entiers avec l'algorithme de tri par insertion.

- La fonction recevra en paramètre le tableau de départ ainsi que le nombre d'éléments.
- Le tri sera fait dans le tableau de départ sans avoir recours à un tableau auxiliaire.

Exercice 5 : Ecrivez un programme qui permet à l'utilisateur de supprimer une valeur d'un tableau préalablement saisi. L'utilisateur donnera l'indice de la valeur qu'il souhaite supprimer. Attention, il ne s'agit pas de remettre une valeur à zéro, mais bel et bien de la supprimer du tableau lui-même ! Si le tableau de départ écrit 12 - 8 - 4 - 45 - 64 - 9 - 2 7 et que l'utilisateur souhaite supprimer la valeur d'indice 4, le nouveau tableau sera : 12 - 8 - 4 - 45 - 9 - 2 - 7.

Exercice 6 : On veut monter une application de gestion de notes des stagiaires, pour ce faire on fait appel à deux fichiers :

Fichiers stagiaires : avec organisation indexée, Accès direct sur la clé code, ouvert en lecture.

Structure :

Code : entier

Nom : 10 Caractères

Prenom : 10 Caractères

Section : 6 Caractères

Fichier Module : avec organisation séquentielle, Accès séquentiel, ouvert en écriture.

Structure :

Code : entier

CodeModule : 20 caractères

NoteMoyenne : Numérique décimal

Travail à faire :

1. Prévoir une solution de saisie des notes pour chaque stagiaire
2. Donner la liste des stagiaires saisis
3. Donner la liste des modules
4. Donner le nombre de stagiaires dans le fichier
5. Calculer la moyenne globale des notes
6. Prévoir un traitement des exceptions pour les contraintes qui posent problèmes au système telles que l'inexistence d'un stagiaire dans le fichier stagiaire ou dans un module dans le fichier Modules ou d'une note.

Exercice 7 : Gestion des ventes d'un magasin. On veut saisir le prix HT d'articles au regard de la TVA (livres, articles spéciaux ou autres articles) et calculer leur prix TTC, connaissant le taux de TVA associé à chaque type (livres : 5,5 %, disque : 20,6 %, articles spéciaux : 33,5 %, autres : 20,6 %).

- 1- Ecrire le programme qui permet de faire le bilan du nombre d'articles vendus et des montants HT et TTC totaux et moyennes.
- 2- On suppose maintenant que chaque taux de TVA est associé à un code connu de l'utilisateur (ex. livres = 1, disque = 2, etc.). Améliorer les structures de données pour simplifier le programme précédent.

Exercice 8 : On dispose d'un tableau de nombres réels positifs T1 de taille N. Ecrire l'algorithme qui permet de trier les valeurs contenus dans T1 dans un second tableau T2, de la plus grande à la plus petite.

Exercice 9 : Faire l'algorithme pour calculer $K = 1 * 1/2 * 1/3 * \dots * 1/n$ ($n > 0$).

Ecrire le programme C permettant de calculer K. N'utilisez pas la classe Clavier.

Exercice 10 : Ecrire un algorithme qui calcule la racine carrée d'un nombre réel : à l'aide de la formule itérative suivante :

$$\begin{cases} X_1 = 1 \\ X_n = \frac{1}{2} \left(X_{n-1} + \frac{A}{X_{n-1}} \right) \end{cases}$$

Exercice 11 : On dispose de toute la monnaie nécessaire en billets de 200 Dhs, 100 Dhs, 50 Dhs, 20 Dhs, et en pièces de 10 Dhs, 5 Dhs, 2 Dhs et 1 Dhs.

Ecrire un programme qui décompose une somme d'argent saisie au Clavier en billets et pièces (en utilisant le plus petit nombre de billets et de pièces possible) et affiche la décomposition.

Exemple :

Une somme d'argent saisie : 38 Dhs sera décomposée comme suit : Un billet de 20 Dhs, Une pièce de 10 Dhs, Une pièce de 5 Dhs, Une pièce de 2 Dhs et Une pièce de 1 Dhs.

Session Juin 2008

Exercice 12 : Soit x une donnée de type réel

$$e^x = 1 + \frac{x^1}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots + \frac{x^n}{n!}$$

Cela est vrai lorsque n tend vers $+\infty$. En informatique, il n'y a pas de notion d'infini. Le terme $(x^n / n!)$ va tendre vers zéro. En informatique, et à partir d'une valeur de n (qui dépend de x), ce terme va être nul.

Réaliser une fonction qui fait l'approximation de l'exponentiel.

Exercice 13 : Que produit l'algorithme suivant ?

Tableau Suite(7) en Entier

Variable i en Entier

Début

Suite(0) ← 1

Suite(1) ← 1

Pour i ← 2 à 7

Suite(i) ← Suite(i - 1) + Suite(i - 2)

i suivant

Pour i ← 0 à 7

Ecrire Suite(i)

i suivant

Fin

Exercice 14 : Ecrivez un algorithme qui calcule le schtroumpf des deux tableaux. Pour calculer le schtroumpf, il faut multiplier chaque élément du tableau 1 par chaque élément du tableau 2, et additionner le tout.

Exercice 15 : Soit une classe de 20 stagiaires. Chaque stagiaire est représenté par les informations suivantes :

Nom	chaîne
Prenom	chaîne
Notes	Tableau réel
Moyenne	réel
Classement	entier

Utilisez un tableau pour contenir les données des stagiaires.

On veut réaliser les traitements suivants :

- Saisir les données nécessaires
- Calculer la moyenne pour chaque stagiaire
- Trier les stagiaires par la moyenne et dans le sens décroissant
- Déterminer le classement pour chaque stagiaire
- Afficher les données de tous les stagiaires.

Exercice 16 : Ecrire un programme permettant, à l'utilisateur de saisir les notes d'une classe. Le programme, une fois la saisie terminée, renvoie le nombre de ces notes supérieures à la moyenne de la classe.

NB : Ne pas oublier des structures de contrôle pour la saisie des notes. La note doit être numérique, comprise entre 0 et 20.

Exercice 17 : Ecrire un programme permettant, à l'utilisateur de saisir les notes d'une classe. Le programme, une fois la saisie terminée, affiche la liste des étudiants ayant une note inférieure à 10. La liste des étudiants doit être triée par ordre alphabétique.

NB : Ne pas oublier des structures de contrôle pour la saisie des notes. La note doit être numérique, comprise entre 0 et 20.

Exercice 18 : Ecrire un programme permettant, à l'utilisateur de saisir les notes d'une classe. Le programme, une fois la saisie terminée, renvoie le nombre de ces notes inférieures à la moyenne de la classe.

NB : Ne pas oublier des structures de contrôle pour la saisie des notes. La note doit être numérique, comprise entre 0 et 40.

Exercice 19 : Faire un programme permettant d'éditer une facture pour les clients. Les produits sont à définir avec leurs prix dans un tableau.

Une fois le client choisi les produits, le programme calculera la somme Hors Taxe en fonction de la quantité de chaque produit et le montant total TTC à payer en ajoutant 20 % de la somme hors taxe.

Le programme doit :

- Demander le nom du client avant de commencer d'éditer la facture.
- Contrôler le prix du produit (>0 Dh)

- Ajouter autant de produits
- Définir un menu de démarrage d'ajout de produits et calcul de la somme à payer.

Exercice 20 : Ecrivez un algorithme permettant à l'utilisateur de saisir un nombre quelconque de valeurs, qui devront être stockées dans un tableau. L'utilisateur doit donc commencer par entrer le nombre de valeurs qu'il compte saisir. Il effectuera ensuite cette saisie. Enfin, une fois la saisie terminée, le programme affichera le nombre de valeurs négatives et le nombre de valeurs positives.

Exercice 21 : A étant un tableau de n réels. On se propose d'écrire une fonction intitulée MaxA renvoyant la valeur maximale dans un tableau.

Exercice 22 : Ecrire un programme qui affiche un triangle isocèle formé d'étoiles de N lignes (N est fourni au clavier) :
Exemple : Nombre de ligne : 8

```

      *
     ***
    *****
   *********
  ***********
 *****
*****
*****

```

Exercice 23 : Ecrire la fonction ADD_MATRICE qui effectue l'addition de deux matrices.

Exercice 24 : Ecrire un programme qui lit une chaîne de caractères et qui convertit toutes les majuscules dans des minuscules et vice-versa.

Exercice 25 : Ecrire un programme qui lit une chaîne de caractère ne dépassant pas 200 caractères et affiche ensuite, la longueur L de la chaîne et le nombre de 'e' existant.

Exercice 26 : Ecrire un programme qui lit N nombres entiers et qui affiche leur somme, leur produit et leur moyenne. Choisir un type approprié pour les valeurs à afficher. Le nombre N est à entrer au clavier.

Exercice 27 : m personnes sont assises autour d'une table circulaire et entament le jeu suivant. Elles fixent entre-elles un entier $k > 0$ et une personne P_1 par laquelle elles commenceront le jeu. On commence à compter de 1 à k en partant de P_1 et en tournant à droite. La $K^{\text{ième}}$ personne quitte la table et on continue à compter de 1 à k à partir de la personne suivante encore à table. La personne gagnante est la dernière qui reste assise. Analyser ce problème et faire un programme permettant de réaliser ce jeu.

Exercice 28 : On se propose d'écrire un programme qui remplit au hasard un tableau de n réels puis les affiche à l'écran. Le programme doit calculer et afficher la valeur moyenne et la valeur Max. les réels doivent être compris entre 0 et 20.

Exercice 29 : Ecrire un programme qui recherche la valeur minimum d'un tableau d'entiers.

Exercice 30 : Ecrire un programme qui trie un tableau de n réels par ordre décroissant.

Exercice 31 : Ecrire un programme qui transfère une matrice M à deux dimensions L et C (dimensions maximales : 10 lignes et 10 colonnes) dans un tableau V à une dimension L*C.

Exemple :

$$\begin{pmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \end{pmatrix} \rightarrow (a \ b \ c \ d \ e \ f \ g \ h \ i \ j \ k \ l)$$

Exercice 32 : Ecrire un programme qui effectue la transposition tA d'une matrice A de dimension N et M en une matrice de dimensions M et N.

Rappel :

$$tA = t \begin{pmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \end{pmatrix} = \begin{pmatrix} a & e & i \\ b & f & j \\ c & g & k \\ d & h & l \end{pmatrix}$$

Exercice 33 : Ecrire un programme qui réalise l'addition de deux matrices A et B de mêmes dimensions N et M.

Rappel:

$$\begin{pmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \end{pmatrix} + \begin{pmatrix} a' & b' & c' & d' \\ e' & f' & g' & h' \\ i' & j' & k' & l' \end{pmatrix} = \begin{pmatrix} a+a' & b+b' & c+c' & d+d' \\ e+e' & f+f' & g+g' & h+h' \\ i+i' & j+j' & k+k' & l+l' \end{pmatrix}$$

Exercice 34 : Faire un programme permettant de calculer la somme des N premiers termes de la série harmonique :
 $1 + 1/2 + 1/3 + \dots + 1/N$

Exercice 35 : Faire un programme permettant de calculer d'afficher la table des produits pour N variant de 1 à 10 :

X*Y	0	1	2	3	4	5	6	7	8	9	10
0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8	9	10
2	0	2	4	6	8	10	12	14	16	18	20
3	0	3	6	9	12	15	18	21	24	27	30
4	0	4	8	12	16	20	24	28	32	36	40
5	0	5	10	15	20	25	30	35	40	45	50
6	0	6	12	18	24	30	36	42	48	54	60
7	0	7	14	21	28	35	42	49	56	63	70
8	0	8	16	24	32	40	48	56	64	72	80
9	0	9	18	27	36	45	54	63	72	81	90
10	0	10	20	30	40	50	60	70	80	90	100

Exercice 36 : Faire un programme qui lit N nombres entiers au clavier et qui affiche leur somme, leur produit et leur moyenne. Choisissez un type approprié pour les valeurs à afficher. Le nombre N est à entrer au clavier.

Exercice 37 : Faire un programme qui calcule le produit scalaire de deux vecteurs d'entiers U et V (de même dimension).

Exemple :

$$(3 \ 2 \ -4) * (2 \ -3 \ 5) = 3 * 2 + 2 * (-3) + (-4) * 5 = -20$$

Exercice 38 : On dispose de deux tableaux A et B (de dimensions respectives N et M), triés par ordre croissant. Fusionner les éléments de A et B dans un troisième tableau FUS trié par ordre croissant.

Exercice 39 : Faire un programme qui construit le triangle de PASCAL de degré N et le mémorise dans une matrice carrée P de dimension N+1.

Exemple :

1							
1	1						
1	2	1					
1	3	3	1				
1	4	6	4	1			
1	5	10	10	5	1		
1	6	15	20	15	6	1	

Exercice 40 : Faire un programme ayant cet effet : saisir deux vecteurs de dimension N puis tester s'il sont égaux ou non. Deux vecteurs sont égaux s'ils ont pour chaque indice la même valeur.

Exercice 41 : Faire un programme qui calcule le produit scalaire de deux vecteurs d'entiers U et V (de même dimension).

Exemple :

$$(3 \ 2 \ -4) * (2 \ -3 \ 5) = 3 * 2 + 2 * (-3) + (-4) * 5 = -20$$

Exercice 42 : On dispose de deux tableaux A et B (de dimensions respectives N et M), triés par ordre croissant. Fusionner les éléments de A et B dans un troisième tableau FUS trié par ordre croissant.

Exercice 43 : Faire un programme qui lit N nombres entiers au clavier et qui affiche leur somme, leur produit et leur moyenne. Choisissez un type approprié pour les valeurs à afficher. Le nombre N est à entrer au clavier.

Exercice 44 : Ecrire un programme qui saisit une chaîne pouvant contenir des espaces et qui affiche chaque mot de la chaîne, le séparateur étant l'espace.

Exemple, on tape : je pense donc je suis

Le programme affiche :

Mot 1 : je
Mot 2 : pense
Mot 3 : donc
Mot 4 : je
Mot 5 : suis

Exercice 45 : Ecrire un programme qui demande à l'utilisateur de taper le prix d'un kilo de tomates, le nombre de kilos de tomates achetés, le taux de tva. Le programme affichera le prix TTC des marchandises.

Exercice 46 : Ecrire une fonction f ayant en paramètres un tableau t1 de taille quelconque et un entier n indiquant la taille du tableau, ainsi qu'un tableau t2 de la même taille que t1. f doit renvoyer par un return un entier nb indiquant le nombre de valeurs comprises entre 0 et 10 dans le tableau t1. f doit mettre dans le tableau t2 les différentes valeurs comprises entre 0 et 10 qu'il a rencontrées dans le tableau t1.

Exercice 47 : Ecrire un programme de recherche de la valeur maximale et minimale d'un tableau [N][M] de réels de taille N x M.

Exercice 48 : On considère une séquence d'entiers s de longueur L représentée dans un tableau T d'entiers défini sur l'intervalle [1...Lmax], $0 < L < Lmax$.

On veut écrire un programme qui remplace dans T la suite s par la suite s' de longueur L' (avec $L' \leq L$), déduite de s en supprimant tous les éléments redondants. Un élément de s est redondant s'il est égal à un autre élément de s. le

programme ne doit pas utiliser de tableau intermédiaire pour créer s'. l'ordre des éléments reste celui de la séquence de départ. Etudier tout d'abord le problème en supposant que T peut ne pas être trié en ordre croissant.

Exercice 49 : Ecrire un programme qui lit le fichier VotreNom.txt crée dans votre dossier, et affiche son contenu. Le programme doit vérifier l'existence de ce fichier.

Exercice 50 : Faire un programme permettant d'écrire dans ce fichier, le programme ne doit pas sauvegarder dans le fichiers des données null.

Exercice 51 : Ecrire un programme qui demande à l'utilisateur de taper 10 entiers entre 0 et 20 qui seront stockés dans un tableau et qui affiche le nombre de fois qu'on a tapé un 0, le nombre de 1, le nombre de 2, ... , le nombre de 20.

Exercice 52 : Considérant un tableau numérique « T » de N éléments, et un deuxième tableau numérique « V » de M éléments. Concevoir un programme qui permet de poser les éléments des deux tableaux dans un troisième tableau numérique « R » de telle façon à l'avoir trié en ordre croissant.

Session Juin 2010

Exercice 53 : On souhaite écrire un algorithme qui permet de convertir un nombre naturel entré par l'utilisateur en binaire.

- Ecrire une fonction qui retourne le nombre de bits nécessaires pour présenter un entier naturel n en binaire.
- Ecrire un algorithme qui permet d'afficher la représentation binaire d'un entier naturel n entré par l'utilisateur.

Exercice 54 : Ecrire un algorithme qui permet de chercher toutes les occurrences d'une valeur donnée dans un tableau de N éléments.

Exemple :

12	10	12	15	14	12	6	12	7	8
----	----	----	----	----	----	---	----	---	---

1^{ère} Occurrence

2^{ème} Occurrence

3^{ème} Occurrence

Dernière Occurrence

Exercice 55 : On donne en entrée un tableau de N éléments de type entier. Donner le programme qui affiche les éléments du tableau qui possèdent leur carrés ($t(i) * t(i)$) dans le même tableau. Les éléments sont rangés dans un ordre aléatoire à l'intérieur du tableau.

Exemple :

1	2	4	7	11	5	9	16	25
---	---	---	---	----	---	---	----	----

Les éléments dont les carés sont présents : **1, 2, 4, 5**

Exercice 56 : Pour un entier n strictement positif on associe $n/2$ si n est pair et $3n + 1$ si n est impair. En réappliquant cette transformation à l'entier obtenu, on définit un algorithme dit de Syracuse. On admettra que pour tout entier strictement positif de départ on finisse toujours par arriver à 1.

On demande d'écrire un programme qui, pour une valeur de départ proposée par l'utilisateur, affiche la liste des entiers obtenus jusqu'à 1, ainsi que le nombre de fois qu'il est nécessaire d'appliquer la transformation pour y arriver.

Voici un exemple de déroulement de cet algorithme :

Valeur e départ (entier strictement positif) ? 12

6 3 10 5 16 8 4 2 1

On doit appliquer 9 fois la transformation avant d'arriver à 1.

Exercice 57 : Quelle est la valeur de s après exécution des instructions suivantes.

```
int i ;
int s = 0 ;
for (i=1 ; i < 6 ; i=i+2)
    s = s + i;
```

- 7
- 9
- 11
- Erreur

Exercice 58 : Ecrire un programme qui transfère une matrice M à deux dimensions L et C dans un tableau V à une seule dimension.

Exercice 59 : La direction d'un supermarché a décidé d'accorder des réductions à ses clients selon le montant d'achat. La réduction est calculée selon les règles suivantes :

- 20% pour un montant d'achat de plus de 5000 Dhs
- 15% pour un montant d'achat entre 3000 Dhs < Montant achat ≤ 5000 Dhs
- 10% pour un montant d'achat entre 1000 Dhs < Montant achat ≤ 3000 Dhs
- Aucune réduction pour un montant d'achat inférieur à 1000 Dhs.

Ecrire un programme qui permet de calculer et d'afficher la réduction et le montant à payer.

Exercice 60 : La direction d'une entreprise désire automatiser le calcul de l'indemnité à verser aux cadres en cas de licenciement.

Après un an d'ancienneté dans l'entreprise, il sera alloué aux cadres licenciés une indemnité tenant compte de leur ancienneté et s'établissant comme suite :

- La moitié du salaire d'un mois par année d'ancienneté : pour la tranche d'ancienneté entre 1 et 10 ans.
- Au-delà de 10 ans un mois de salaire par année d'ancienneté.
- Une indemnité supplémentaire serait allouée aux cadres âgés de plus de 45 ans de :
 - o 2 mois si le cadre est âgé de 46 à 49 ans.
 - o 5 mois si le cadre est âgé de plus de 50 ans.

Ecrire un programme qui permet de saisir l'âge, l'ancienneté et le dernier salaire et d'afficher l'indemnité du cadre.

Exercice 61 : Ecrire un algorithme qui effectue la lecture d'une matrice carrée A ainsi que sa taille n et affiche la matrice transposée tA de A (Pour une matrice A(i, j), tA(j, i)).

Exercice 62 : Ecrire un programme permettant de trier un tableau de n réels dans un ordre croissant en utilisant la méthode de tri bulle.

Exercice 63 : Ecrire un algorithme qui effectue la lecture d'une matrice carrée A ainsi que sa taille n et affiche la trace de A. (Pour une matrice A(i,j), $\text{trace}(A) = \sum a_{i,i}$ la somme des éléments sur la diagonale.

Exercice 64 : Ecrire un programme qui demande à l'utilisateur de saisir un entier N et qui affiche la figure suivante sous forme des étoiles.

```
N = 3
***
**
*
```

Exercice 65 : Ecrire un programme permettant de saisir le prix unitaire et la quantité commandée d'un article. Le programme affichera le prix à payer, le port, et la remise sachant que :

- Le port est gratuit si le montant hors taxe est supérieur à 1000 Dhs
- Le port est 3% dans le cas contraire.
- La remise est de 5% si le montant hors taxe est compris entre 300 et 1000 Dhs et de 10% au-delà de 1000 Dhs.

NB : Montant hors taxe = prix unitaire x quantité commandée.

Exercice 66 : Ecrire un programme qui permet de comparer et de calculer la somme de deux nombres. Pour ce faire ce menu doit être affiché.

Entrez le nombre correspondant à votre choix

1. Comparer deux entiers
2. Somme de deux entiers
3. Quitter

Exercice 67 : On modélise un nombre entier de la base octale par un tableau d'entiers de dimension 10. Chaque élément du tableau contient un chiffre de 0 à 7. Le chiffre du poids faible occupera la case d'indice 9 du tableau.

Exemple : 3406

Le nombre est saisi en entrant successivement les chiffres du poids plus faible au poids plus fort, on marque la fin par un chiffre qui n'est pas compris entre 0 et 7.

Exercice 68 : Soit un tableau de 100 valeurs ordonnées de manière croissante. Ecrire un programme permettant de créer deux nouveaux tableaux ordonnés de manière croissante, contenant l'un les valeurs paires, l'autre les valeurs impaires.

Exemple :

Tableau initial

2	4	10	13	15	21	28	55	44	77
---	---	----	----	----	----	----	----	----	----

Tableau des valeurs paires

2	4	10	28	44
---	---	----	----	----

Tableau des valeurs impaires

13	15	21	55	77
----	----	----	----	----

Exercice 69 : Dans un magasin les caissières saisissent pour chaque produit :

- Le prix unitaire
- La quantité achetée
- La catégorie du produit

Il y a une taxe pour chaque catégorie du produit

Catégorie	Taxe
1	6%
2	9%
3	12%
4	15%
5	20%

Ecrire un programme qui permet d'afficher le prix à payer pour un client.

Exercice 70 : Ecrire un programme qui demande un entier N positif, et deux entiers Bd et Ba (compris entre 2 et 10) et convertit N de la base Bd en Base Ba.

Exercice 71 : Un palindrome est une chaîne de caractères qui se lit de gauche à droite ou de droite à gauche et donne le même résultat.

Exemple : « radar », « laval ».

Ecrire un programme qui permet de tester si un mot est un palindrome ou non.

Exercice 72 : Ecrire un programme permettant de modéliser un nombre entier binaire par un tableau d'entiers de dimension 10. Chaque élément du tableau contient un bit 0 ou 1. Le bit du poids faible occupera la case d'indice 9 du tableau.

Exemple : 10010

0	0	0	0	0	1	0	0	1	0
---	---	---	---	---	---	---	---	---	---

Le nombre est saisi en entrant successivement les bits du poids plus faible au poids plus fort, on marque la fin par un chiffre de 0 et 1.

Exercice 73 : Ecrire un programme qui permet d'afficher tous les nombres abondants inférieurs à 50. Un nombre abondant est un nombre entier naturel n qui est strictement inférieur à la somme de ses diviseurs stricts.

Exemple : 12 est abondant car $1 + 2 + 3 + 4 + 6 > 12$

Exercice 74 : Ecrire un programme qui lit un entier X et un tableau A du type int au clavier et élimine toutes les occurrences de X dans A en tassant les éléments restants (décalage).

Exercice 75 : Faire un programme pour le calcul et l'affichage suivant :

1 * 8 + 1 = 9
12 * 8 + 2 = 98
123 * 8 + 3 = 987
1234 * 8 + 4 = 9876
12345 * 8 + 5 = 98765
123456 * 8 + 6 = 987654
1234567 * 8 + 7 = 9876543
12345678 * 8 + 8 = 98765432
123456789 * 8 + 9 = 987654321

Exercice 76 : Ecrire un programme permettant de calculer le Poids Idéal (PI) obtenu par la formule de Lorentz :

Pour les femmes : $PI = t - 100 - [(t - 150) / 2]$

Pour les hommes : $PI = t - 100 - [(t - 150) / 4]$

T = taille en centimètre

Exercice 77 : Ecrire un programme qui permet d'insérer une valeur X dans un tableau T, supposé trié, de façon à respecter l'ordre des éléments de T. le tableau T contient N éléments et sera dimensionné à N + 1 (pour permettre de ranger X)

Si N=10 et T =

17	17	21	23	24	26	27	30	30	38
----	----	----	----	----	----	----	----	----	----

Si X = 25 On doit obtenir :

17	17	21	23	24	25	26	27	30	30	38
----	----	----	----	----	----	----	----	----	----	----

Exercice 78 : Ecrire un programme de recherche de la valeur maximale d'une matrice de réels de taille n x m.

Exercice 79 : Faire un programme pour le calcul et l'affichage suivant :

9 * 9 + 7 = 88
98 * 9 + 6 = 888
987 * 9 + 5 = 8888
9876 * 9 + 4 = 88888
98765 * 9 + 3 = 888888
987654 * 9 + 2 = 8888888
9876543 * 9 + 1 = 88888888
98765432 * 9 + 0 = 888888888

Exercice 80 : Ecrire un programme de recherche de la valeur maximale d'une matrice de réels de taille n x m.

Exercice 81 : Un nombre parfait est un entier positif supérieur à 1, égal à la somme de ses diviseurs.

NB : On compte 1 comme diviseur, mais on ne compte pas comme diviseur le nombre lui-même.

Exemple :

6 est un nombre parfait puisque $6 = 3 + 2 + 1$.

1. Ecrire une fonction qui prend pour argument un nombre entier n et retourne Vrai si n est parfait ou Faux sinon.
2. Ecrire un programme permettant de lister les 500 premiers nombres parfaits.

Exercice 82 : Donner le schéma itératif « Tant que » ; et expliquer son fonctionnement.

Exercice 83 : Pour chacune des deux séquences algorithmiques suivantes, dire s'il est possible de simplifier son écriture. Si oui, comment ; et sinon pourquoi ?

<p>a-</p> <p>Si A>B Alors</p> <p style="padding-left: 20px;">C ← 2 * D</p> <p style="padding-left: 20px;">D ← A - B</p> <p style="padding-left: 20px;">A ← B + D</p> <p style="padding-left: 20px;">B ← A + D</p> <p>Sinon</p> <p style="padding-left: 20px;">C ← 2 * D</p> <p style="padding-left: 20px;">D ← B - A</p> <p style="padding-left: 20px;">A ← B + D</p> <p style="padding-left: 20px;">B ← A + D</p> <p>FinSi</p>	<p>b-</p> <p>Si U = 5 Alors</p> <p style="padding-left: 20px;">U ← 2 * V</p> <p style="padding-left: 20px;">V ← V + U</p> <p style="padding-left: 20px;">W ← V</p> <p>Sinon</p> <p style="padding-left: 20px;">U ← 2 * V</p> <p style="padding-left: 20px;">V ← V + U</p> <p style="padding-left: 20px;">W ← -V</p> <p>FinSi</p>
--	--

Exercice 84 : Une matrice carrée est un tableau à deux dimensions dont le nombre de lignes est égale au nombre de colonnes. Une matrice est dite équilibrée si la somme des éléments de chaque ligne est égale à zéro, la somme des éléments de chaque colonne est égale à zéro et la somme des éléments des deux diagonales est égale à zéro aussi. Ecrire un algorithme qui lit une matrice carrée d'entiers et vérifie si elle est équilibrée ou non. Vous devez afficher soit la mention « **Votre matrice est équilibrée** » ou la mention « **Votre matrice est non équilibrée** ».

Exercice 85 : Ecrire un programme qui permet de trouver les nombres premiers inférieurs ou égaux un entier N donné par l'utilisateur.

La méthode consiste à remplir un tableau, dans l'ordre du plus petit au plus grand, par les entiers compris entre 2 et N (N étant l'entier limite précisé par l'utilisateur) ; puis de parcourir le tableau depuis le début, en gardant un entier rencontré et éliminant tous ses multiples, par leur mise à zéro par exemple. A la fin du parcours, tous les entiers restants (non nuls) sont des nombres premiers.

Exercice 86 : Ecrire un programme permettant de compter le nombre de diviseurs d'un nombre entier donné par l'utilisateur, puis les afficher du plus grand au plus petit.

Exemple d'exécution :

Le nombre de diviseurs de 8 est : 4
Les diviseurs de 8 par ordre décroissant : 8, 4, 2, 1

Exercice 87 : Ecrire un programme permettant de saisir un tableau de 10 valeurs entières et de diviser l'ensemble des éléments du tableau par la valeur du n^{ième} élément du tableau, n étant saisi par l'utilisateur.

Exemple d'exécution :

Soit le tableau saisi : 8 - 4 - 6 - 2 - 4 - 12 - 6 - 14 - 10 - 8
N = 4 :
Tableau résultat : 4 - 2 - 3 - 1 - 2 - 6 - 3 - 7 - 5 - 4

Exercice 88 : Ecrire une fonction nommée « puissance(x,y) » permettant de retourner x^y, avec x et y des réels passés en paramètres.

Ecrire un programme qui utilise la fonction « puissance » de la question précédente pour calculer l'équivalent décimal d'un nombre binaire.

Rappel : $(1011101)_2 = 1X2^0 + 0X2^1 + 1X2^2 + 1X2^3 + 0X2^4 + 1X2^5 = (45)_{10}$

Exemple d'exécution :

Entrez le nombre binaire : 101101
L'équivalent décimal est : 45

Exercice 89 : Ecrire un programme permettant de compter le nombre de multiples (inférieurs à 1000) d'un nombre entier donné par l'utilisateur, puis les afficher du plus grand au plus petit.

Exemple d'exécution :

Le nombre de multiples (<1000) de 120 est : 8
Les multiples de 120 par ordre décroissant :
960 - 840 - 720 - 600 - 480 - 360 - 240 - 120

Exercice 90 : Ecrire un programme permettant de saisir un tableau de 10 valeurs entières et de multiplier l'ensemble des éléments du tableau par la valeur du n^{ième} élément du tableau, n étant saisi par l'utilisateur.

Exemple d'exécution :

Soit le tableau saisi : 8 - 4 - 6 - 2 - 4 - 12 - 6 - 14 - 10 - 8
N = 4 :
Tableau résultat : 4 - 2 - 3 - 1 - 2 - 6 - 3 - 7 - 5 - 4

Exercice 91 : Ecrire une fonction nommée « puissance(x,y) » permettant de retourner x^y , avec x et y des réels passés en paramètres.
Ecrire un programme qui utilise la fonction « puissance » de la question précédente pour calculer l'équivalent décimal d'un nombre octal.

Rappel : $(177)_8 = 1X8^0 + 0X8^1 + 1X8^2 + 1X8^3 + 0X8^4 + 1X8^5 = (127)_{10}$

Exemple d'exécution :

<p>Entrez le nombre octal : 177 L'équivalent décimal est : 127</p>	➡
--	---

Exercice 92 : Ecrire un programme permettant de compter le nombre de puissances successives (inférieurs à 1000) d'un nombre entier donné par l'utilisateur, puis de les afficher du plus grand au plus petit.

Rappel : Les puissances successives de 2 sont : $2^0 = 1, 2^1 = 2, 2^2 = 4, 2^3 = 8, 2^4 = 16, 2^5 = 32, 2^6 = 64, 2^7 = 128 \dots$

Exemple d'exécution :

<p>Le nombre de puissances successives (<1000) de 5 est : 5 Les puissances successives de 5 par ordre décroissant : 625, 125, 25, 5, 1</p>

Exercice 93 : Ecrire un programme permettant de saisir un tableau de 10 valeurs entières et d'ajouter à l'ensemble des éléments du tableau par la valeur du $n^{ième}$ élément du tableau, n étant saisi par l'utilisateur.

Exemple d'exécution :

<p>Soit le tableau saisi : 8 - 4 - 6 - 2 - 4 - 12 - 6 - 14 - 10 - 8 N = 5 : Tableau résultat : 15 - 11 - 13 - 10 - 14 - 19 - 13 - 21 - 17 - 15</p>

Exercice 94 : Ecrire une fonction nommée « puissance(x,y) » permettant de retourner x^y , avec x et y des réels passés en paramètres.

Ecrire un programme qui utilise la fonction « puissance » de la question précédente pour calculer la somme : $1/x^1 + 1/x^2 + \dots + 1/x^n$ pour x réel donné et n entier donné saisi au clavier.

Exemple d'exécution :

<p>Entrez la valeur de x : 3 Entrez la valeur de n : 5 La somme est : 1.497942</p>	↻
---	---

Exercice 95 : Ecrire un programme qui permet d'entrer un tableau de n valeurs numériques ; et de les afficher sur deux colonnes, de manière de retrouver ces valeurs dans leur ordre initial, si on les lit colonne par colonne.

Exemples d'exécution :

<pre> *** Répartition d'une suite de nombres sur deux colonnes *** Donnez le nombre de valeurs : 7 Donnez les valeurs : 14 12 25 18 22 17 11 Votre suite est : (à lire colonne par colonne) 14 - 22 12 - 17 25 - 11 18 </pre>

NB. Chaque ligne est affichée par une même et seule instruction de sortie.

Exercice 96 : Ecrire un programme qui permet de saisir et de trier les éléments (entiers) d'un tableau au fur et à mesure de la saisie de ses éléments (c'est-à-dire, le programme permettra au début de son exécution la saisie de la taille et des valeurs du tableau, en insérant chaque valeur reçue, de manière à ce qu'on fin de la saisie, le tableau soit trié, par ordre croissant. Les tableaux intermédiaires triés doivent être affichés au fur et à mesure de la saisie des valeurs).

Exemple d'exécution :

<pre> *** Recherche dans un tableau *** Donnez le nombre de valeur de votre tableau : 6 Donnez une valeur de votre tableau : 4 Valeurs du tableau intermédiaire : 4 Donnez une valeur de votre tableau : </pre>

12
 Valeurs du tableau intermédiaire : 4, 12
 Donnez une valeur de votre tableau :
 5
 Valeurs du tableau intermédiaire : 4, 5, 12
 Donnez une valeur de votre tableau :
 8
 Valeurs du tableau intermédiaire : 4, 5, 8, 12
 Donnez une valeur de votre tableau :
 2
 Valeurs du tableau intermédiaire : 2, 4, 5, 8, 12
 Donnez une valeur de votre tableau :
 17
 Valeurs du tableau intermédiaire : 2, 4, 5, 8, 12, 17

Votre tableau est constitué, dans l'ordre, des valeurs :
 2, 4, 5, 8, 12, 17

Exercice 97 : Ecrire un programme qui permet de chercher et afficher tous les entiers « cubiques » ayant trois chiffres. Un entier naturel de trois chiffres (compris entre 100 et 999) est cubique s'il est égal à la somme des cubes de trois chiffres.

Exemple : 153 est cubique puisque $153 = 1^3 + 5^3 + 3^3$

Exercice 98 : Ecrire un programme qui permet d'entrer un tableau de n valeurs numériques ; et de les afficher sur deux colonnes, de manière de retrouver ces valeurs dans leur ordre initial, si on les lit ligne par ligne.

Exemples d'exécution :

```

*** Répartition d'une suite de nombres sur deux colonnes ***

Donnez le nombre de valeurs : 7
Donnez les valeurs :
14
12
25
18
22
17
11

Votre suite est : (à lire colonne par colonne)
14   -   12
25   -   18
22   -   17
11
  
```

NB. Chaque ligne est affichée par une même et seule instruction de sortie.

Exercice 99 : Ecrire une fonction qui calcule le factoriel d'un entier n donné (le factoriel d'un nombre n s'écrit n! et $n! = 1 \times 2 \times 3 \times \dots \times n$. Exemple : $3! = 1 \times 2 \times 3 = 6$).

Ecrire une fonction qui permet de calculer le nombre de combinaisons de p parmi n éléments

$$\binom{n}{p}$$

Rappel : $\binom{n}{p} = \frac{n!}{p!(n-p)!}$

Ecrire un programme utilisant ces fonctions, pour les tester.

Exercice 100 : Ecrire un programme qui permet d'entrer un tableau de n valeurs numériques ; puis de chercher et afficher toutes ses valeurs, distincts, avec pour chacune les positions où elle apparaît.

Exemple :

Soit le tableau : 1 - 2 - 3 - 1 - 2 - 10 - 1

Le résultat :

Les positions de 1 sont : 1, 4, 7

Les positions de 2 sont : 2, 5

Les positions de 3 sont : 3

Les positions de 10 sont : 6

Exercice 101 : Ecrire un programme qui permet d'entrer un tableau de n valeurs numériques ; puis d'inter changer ses éléments de manière que la 1^{ère} valeur devienne la dernière, la deuxième devienne l'avant dernière, ... ; et l'afficher, dans ce nouvel ordre.

Exemples d'exécution :

```

*** Inversion d'une suite de nombres ***
  
```

Donnez le nombre de valeurs : 7
Donnez les valeurs :
14
12
25
18
22
17
11

Votre suite devient :
11 - 17 - 22 - 18 - 25 - 12 - 14

NB. Chaque ligne est affichée par une même et seule instruction de sortie.

Exercice 102 : Ecrire une fonction qui calcule le factoriel d'un entier n donné (le factoriel d'un nombre n s'écrit n! et $n! = 1 \times 2 \times 3 \times \dots \times n$. Exemple : $3! = 1 \times 2 \times 3 = 6$).

Ecrire une fonction qui permet de calculer le nombre d'arrangements de p parmi n éléments

$$A_p^n$$

Rappel : $A_p^n = \frac{n!}{(n-p)!}$

Ecrire un programme utilisant ces fonctions, pour les tester.

Exercice 103 : Ecrire un programme qui permet d'entrer un tableau de n valeurs numériques ; puis de compter et afficher le nombre de ses valeurs distinctes.

Exercice 104 : Ecrire un programme qui permet d'entrer deux nombres réels et d'afficher l'une des deux mentions : « Positif » ou « Négatif », suivant le signe de leur produit, sans toutefois le calculer.

Exercice 105 : Ecrire un programme qui simule une caisse enregistreuse. Le programme permettra à l'utilisateur d'entrer une série de prix des achats d'un client ; et arrêter la lecture quand l'utilisateur entre 0. Il affichera ensuite le montant total qui correspond à ces achats.

Le programme permettra par la suite d'entrer la somme d'argents présentée par le client, et de calculer le montant qu'on doit lui remettre, écrit en nombre de billets et pièces qui le constituent, de manière qu'on en utilisera un minimum, soit d'utiliser les billets et pièces dans le sens décroissant de leurs valeurs.

Exemple d'exécution :

```
Montant achats (en Dhs) :
12,20
4,00
6,40
0

Total achats : 22,60 Dh

Montant présenté (en Dh) : 50

Montant à rendre : 27,40 Dh
1 billet de 20 Dh
1 pièce de 5 Dh
1 pièce de 2 Dh
2 pièces de 20 Dh
```

Exercice 106 : Un nombre palindrome est nombre symétrique (qui peut être lu de gauche à droite, comme de droite à gauche) ; soit la forme $c_1 c_2 c_3 \dots | \dots c_3 c_2 c_1$, où les c_i sont les chiffres du nombre. Tous les chiffres 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 sont des palindromes à un chiffre. Il y en a 9 palindromes à deux chiffres : 11, 22, 33, 44, 55, 66, 77, 88, 99.

Ecrire un programme qui calcule et affiche, de la même manière, le nombre des nombres palindromes de trois chiffres ; suivi de ceux de quatre chiffres.

Exercice 107 : Soit la suite numérique définie par $A_0 = 1$; $A_n = nA_{n-1}$ (pour tout $n \geq 1$).

Ecrire un programme qui calcule la valeur numérique du polynôme de degré n :

$$P(x) = A_n x^n + A_{n-1} x^{n-1} + \dots + A_1 x + A_0$$

Pour un couple (x, n) donné par l'utilisateur.

Exercice 108 : Ecrire un programme permettant à l'utilisateur d'entrer une suite de valeurs numériques, indifféremment positifs et négatifs, terminées par un zéro ; puis regrouper les valeurs négatives au début du tableau, et celles qui sont

positives à la suite des valeurs négatives, toujours dans l'ordre où elles étaient entrées pour chaque groupe. Le programme procédera en fin à l'affichage de la série dans ce nouvel ordre.

Exemple :

La série

3	-6	-3	4	2	-15	-2	13
---	----	----	---	---	-----	----	----

deviendra

-6	-3	-15	-2	3	4	2	13
----	----	-----	----	---	---	---	----

Exercice 109 : Ecrire un programme qui inverse l'ordre des chiffres d'un nombre donné par l'utilisateur.
Exemple : le nombre 1234 deviendra 4321.

Exercice 110 : Un nombre parfait est un nombre entier naturel qui égal à la somme de ses diviseurs stricts ; c'est-à-dire ses diviseurs entiers positifs, autres que lui-même.

Le premier nombre parfait est 6, puisque la somme de ses diviseurs stricts 1, 2 et 3 est égale à 6 ($1 + 2 + 3 = 6$).

Ecrire un programme qui cherche tous les nombres parfaits inférieurs à une limite donnée par l'utilisateur.

Exercice 111 : Ecrire un programme qui permet de trouver les couples de nombres amis inférieurs à une limite, donné par l'utilisateur.

Deux nombres sont dits « amis » si la somme des diviseurs stricts de l'un est égale à l'autre et inversement.

C'est le cas, par exemple, de 220 et 284 puisque :

220 a pour diviseurs : 1, 2, 4, 5, 10, 11, 20, 22, 44, 55, 110 (somme=284) ; et 284 a pour diviseurs : 1, 2, 4, 71 et 142 (somme=220).