

**Exercice 1 :**

Pour chacun des cas suivants donner l'algorithme et le code Pascal d'un sous programme qui permet de :

- 1) Saisir un caractère Majuscule.
- 2) Saisir une chaîne de caractère non vide et de longueur maximale égale à 20.
- 3) Vérifier est-ce qu'une chaîne de caractère donnée est alphabétique ou non.
- 4) Remplir un tableau T par N entiers positifs croissant.
- 5) Remplir un tableau T par N caractères Majuscules aléatoires
- 6) Afficher un tableau T de N éléments.
- 7) Compter l'occurrence (nombre d'apparition) d'un caractère dans une chaîne.
- 8) Vérifier la présence d'un caractère dans une chaîne.
- 9) Déterminer le maximum d'un tableau.
- 10) Inverser une chaîne de caractère.

**Exercice 2 :**

Soit la fonction Traitement suivante écrite en Pascal :

```

FUNCTION Traitement (d,f :integer;T:tab): .....;
VAR ..... ;
BEGIN
    Indmin: =d;
    For i: = d+1 to f do
        IF T[i] < T[indmin] THEN
            Begin
                Indmin: = i;
            End;
        Traitement: = indmin;
    END;

```

1. Déterminer et compléter le type de cette fonction ainsi que la partie déclaration des variables locales.
2. Quelle est la valeur renvoyée par la fonction Traitement si **d=2**, **f=5** et le tableau contient les éléments suivants :

T	-10	5	0	-6	10	13
i	1	2	3	4	5	6

3. Quel est le rôle de cette fonction ?

**Exercice 3 :**

Soit l'Algorithme suivant :

```

0) DEF FN Traitement (T : Tab ; N : Entier) : Entier
1) NB ← 0
2) Pour i de 1 à N faire
    Si Non (majus (T [i] dans ["A", "E", "O", "I", "U", "Y"]) alors
        NB ← NB + 1
    Fin si
Fin Pour
3) Traitement ← NB
4) Fin Traitement

```

1. Préciser le rôle de la fonction Traitement suite à l'exécution suivante :

T	"T"	"e"	"c"	"h"	"n"	"o"	"l"	"o"	"g"	"i"	"e"
	1	2	3	4	5	6	7	8	9	10	11

2. Transformer la fonction Traitement en une procédure

**Exercice 4 :**

Soit la fonction booléenne VERIF suivante :

```

FUNCTION VERIF (Ch: string): .....;
Var .....;
    .....;
Begin
    Test: = False;
    Vc: = 0;
    Repeat
        Vc: = Vc+1;
        If Not (Upcase (Ch [Vc] )in ['A'..'Z']) Then
            Begin

```

```

Test: = True;
End;
Until (Test) or (Vc = Length (Ch));
..... ;
End;

```

1. Compléter les pointillés par les données manquantes.
2. Quel est le rôle de cette fonction ?

**Exercice 5 :**

Soit l’algorithme de la fonction suivante :

```

0) DEF FN Traitement (ch : ..... ) : .....
1) P ← pos ( ' ', ch)
2) Nb ← 0
3) Tant que (p<>0) faire
    Efface (ch, p, 1)
    .....
    .....
Fin tant que
4) Traitement ← .....
5) Fin Traitement

```

Cette fonction permet de calculer le nombre des mots dans une phrase ch.

On suppose que :

- o La phrase **ch** ne commence pas par un espace
  - o La phrase **ch** ne se termine pas par un espace
  - o Entre deux mots de la phrase **ch** il y a un espace unique
- a. Compléter les points par le code convenable.
  - b. Transformer la fonction Traitement en une procédure Traitement.

**Exercice 6 :**

```

function inconnu (..... : .....;..... : .....):.....;
var
    .....;
    .....;
begin
    tr:=(1=0) ; i:=0;
    repeat
        i:=i+1;
        tr:=(ch[i]=c);
    until (i=length(ch)) or (tr);
inconnu:=.....;
end;

```

- 1) compléter les pointilles par les données manquantes.
- 2) déterminer le résultat retourné par la fonction pour chacun des cas suivants :
  - a. inconnu ('algorithme', 'g') > .....
  - b. inconnu ( 'pascal', 'H') > .....
  - c. inconnu ( '1H5', '5') > .....
- 3) donner le rôle de la fonction inconnu
- 4) convertir la fonction inconnue en une procédure
- 5) recopier et compléter le tableau suivant sachant que l’appel se fait en utilisant une variable X:

Appel de la fonction inconnu	Appel de la procédure inconnu
.....	.....

**Exercice 7 :**

Ecrire un programme permettant d’afficher tous les couples d’entiers (m,n) vérifiant la propriété suivante :

$M \in [1,100]$  et  $n \in [2,15]$

Et m figure dans l’écriture du produit m.n

Exemples :

- 1- Si m = 20 et n= 6  
Alors le produit m.n = 120 contient le nombre 20 dans ce cas le couple (20,6) sera affiché.
- 2- Si m = 20 et n = 12  
Alors le produit m.n = 20 \* 12 = 240 ne contient pas le nombre 20.

**Exercice 8 :**

Sur les touches (2, 3, 4, 5, 6, 7, 8 et 9) du clavier d'un téléphone portable, sont inscrites des lettres pour écrire des messages en plus des chiffres.

Par exemple, sur la touche 5 sont inscrites les lettre J, K et L.

Pour taper la lettre J on appuie une seule fois.

Pour taper la lettre K on tape deux fois.

Pour taper la lettre L on appuie trois fois.

Ecrire un programme pascal permettant de déterminer et d'afficher le nombre total d'appuies sur les touches du clavier d'un téléphone portable pour saisir un mot donné de N lettres, supposées non accentuées.

( $4 \leq N \leq 9$ ).

Indication :

La figure suivante donne la répartition des lettres sur les touches du clavier d'un téléphone portable.

1	2 ABC	3 DEF
4 GHI	5 JKL	6 MNO
7 PQRS	8 TUV	9 WXYZ
* +	0 -	#

**Exercice 9 :**

Soit l'algorithme suivant :

0) DEF FN Somme (p : entier) : entier

1)  $S \leftarrow 0$

Pour i de 1 à (p div 2) faire

Si (p mod i = 0) Alors

$S \leftarrow S+i$

FinSi

Fin pour

2) Somme  $\leftarrow S$

3) Fin Somme

1. Exécuter cet algorithme pour p=6 puis donner son rôle.
2. Transformer la fonction Somme en une procédure.
3. Ecrire un algorithme d'une procédure Saisie, permettant de saisir deux entiers positifs non nuls.
4. En utilisant les deux modules **Somme** et **saisie**, écrire un algorithme d'un programme intitulé Amis, qui permet de saisir deux entiers positifs non nuls et d'afficher s'ils sont amis ou non. Sachant que deux entiers m et n sont dis amis, si la somme des diviseurs de n est égal à m et inversement.

**Exercice 10 :**

T étant un tableau de dimension N\_MAX (N\_MAX = 100), remplir par N caractères alphabétiques ( $N \leq N\_MAX$ ).

On vous demande d'écrire un programme nommé ECLATEMENT qui permet de:

- ◇ Lire un entier N strictement positif ( $0 < N \leq N\_MAX$ ).
- ◇ Remplir le tableau T.
- ◇ Éclater T en deux tableaux Tc (contenant J élément consonnes de T) et Tv (contenant K élément voyelles de T).
- ◇ Afficher les deux tableaux Tv et Tc.

**Exercice 11 :**

Un nombre est dit parfait s'il est égal à la somme de ces diviseurs ("sauf lui même").

Exemple :  $6=3+2+1$ .

Ecrire une analyse et un algorithme du programme intitulé **parfait** qui permet de saisir un entier strictement positif puis il vérifie s'il est parfait.

**Exercice 12 :**

La « **multiplication Russe** » est une méthode particulière permettant la multiplication de deux entiers A et B en utilisant seulement la multiplication par 2, la division par 2 et l'addition.

**Exemple :** pour A =17 et B = 19, le produit de A par B se fait comme suit :

A    B  
17   19

Le premier nombre est divisé par 2 (division entière) et le deuxième est multiplié par 2 : on aura

8    38

Le processus se répète jusqu'à avoir dans la première colonne 1 :

17   19  
8    38  
4    76  
2    152  
1    304

Le résultat est la somme des nombres de la deuxième colonne qui sont en face des nombres impairs de la première colonne (donc les nombres de la deuxième colonne qui sont en face des nombres pairs de la première colonne seront ignorés).

→	17	19	
	8	38	Ignoré
	4	76	Ignoré
	2	152	Ignoré
→	1	304	
<b>17 * 19 =</b>		<b>19 + 304</b>	<b>= 323</b>

- ❖ On veut écrire un programme qui lit deux entiers positifs non nuls et inférieur à 100, calcule et affiche le produit de deux entiers par la méthode Russe.

**Exercice 13 :**

Soit le programme intitulé **info** qui permet de :

- ✓ Saisir la taille **N** d'un tableau **T**, avec ( $1 < N < 15$ ).
- ✓ Remplir un tableau **T** par **N** chaînes des caractères tel que la taille de chacune est dans [3..20].
- ✓ Chercher et afficher tous les chaînes Totalogramme contenue dans **T**.

« Une chaîne de caractères est dite Totalogramme si elle commence et se termine par la même lettre. » (Sans distinction entre majuscule et minuscule)

**Exemple :** Pour **N=6** :

<b>T</b>	Samir	système	temporairement	Bonjour	ses	elle
	1	2	3	4	5	6

Les mots totalogramme sont : temporairement, ses, elle

**Exercice 14 :**

Un texte est dit tautogramme si tous les mots commencent par la même lettre.

**Exemple :**

« Mazarin, ministre malade, méditait même moribond malicieusement mille maltôtes »

On suppose que la phrase est écrite correctement. (Pas d'espace en double)

Ecrire un programme qui permet de vérifier si une chaîne de caractère saisie au clavier est tautogramme ou non.

**Exercice 15 :**

Ecrire un programme qui permet de dire si deux chaînes ch1 et ch2 (non vides) sont anagrammes ou non.

Deux chaînes sont dites anagrammes si elles sont formées par les mêmes caractères.

Exemple :

« Chien » et « Chine » sont anagrammes.

**Exercice 16 :**

Ecrire un programme qui permet de remplir un tableau **T** par **N** caractères alphabétique aléatoires ( $2 \leq N \leq 10$ ) puis déterminer et afficher le nombre d'occurrence d'un caractère alphabétique donnée dans le tableau **T**.

**Exemple :**

<b>T</b>	A	f	r	A	W	c
----------	---	---	---	---	---	---

- ❖ Si **Car="r"** le programme affiche « le caractère r existe dans le tableau ».
- ❖ Si **Car="a"** le programme affiche « le caractère a n'existe pas dans le tableau ».