



**Exercice n°1 :**(2points):

Soit la procédure **Inconnue** Suivante :

```

0)DEF PROC Inconnue (Var M :mat ;N,i,j :entier)
1) SI  $i \leq n$  alors
    SI  $j \leq i$  alors
        Aux  $\leftarrow$  M[i,j]
        M[i,j]  $\leftarrow$  M[j,i]
        M[j,i]  $\leftarrow$  aux
        Proc Inconnue (M,N, i , j+1)
    Sinon
        Proc Inconnue (M,N, i+1 , 1)
    Fin SI
Fin SI
2)Fin Inconnue
    
```

**Mat** est un type utilisateur représentant un tableau d'entiers de 20 lignes et de 20 colonnes. On donne l'algorithme de la procédure **Inconnue** suivante qui permet de faire la permutation de deux parties de matrice par rapport à la **première diagonale**.

**Exemple :** pour  $n=3, i=1, j=1$ , et **M** =

 <b>1<sup>er</sup> Diagonale</b>	9	6	0		9	4	3
	4	5	2		6	5	7
	3	7	1		0	2	1

Apporter les modifications nécessaires à l'algorithme de cette procédure pour que le permutation sera faire par rapport au **deuxième diagonale**.

**Exercice n°2 : (3 pts)**

L'une des règles de divisibilité par 19 consiste à ajouter à N augmenté de son chiffre des unités le double du chiffre supprimé et recommencer éventuellement avec le nombre ainsi obtenu jusqu'au moment où l'on peut conclure à la divisibilité (**jusqu'à obtenu une valeur  $\leq 19$** ).

Exemple:

N=345686

$$34568 + 2 \cdot 6 = 34580$$

$$3458 + 2 \cdot 0 = 3458$$

$$345 + 2 \cdot 8 = 361$$

$$36 + 2 \cdot 1 = 38$$

$$3 + 2 \cdot 8 = 19 \quad \text{qui est divisible par 19}$$

a) Appliquer le principe décrit précédemment pour le nombre N= **179583**.

b) Proposer l'algorithme d'une fonction qui vérifie si un nombre est divisible par 19 ou non.

### Exercice n°3 : (5 pts)

Pour calculer le  $C_n^p$  c'est à dire la combinaison de p élément parmi n, on peut utiliser :

$$C_n^p = 1 \text{ Quand } p=0$$

$$C_n^p = 1 \text{ Quand } p=n$$

$$C_n^p = C_{n-1}^p + C_{n-1}^{p-1} \text{ dans tout autre cas}$$

#### Question :

1- Ecrire l'analyse d'une fonction **récursive** Comb(n,p) qui détermine la combinaison de p élément parmi n en utilisant la méthode ci-dessus

2- La suite GL est calculée par la formule de **Gohierre de Longchanps** :

$$GL = 1 + 2 \frac{1}{3C_2^1} + 2^2 \frac{1}{5C_4^2} + \dots + 2^n \frac{1}{(2n+1)C_{2n}^n}$$

En utilisant la fonction **Comb(n,p)**, écrire l'algorithme d'un module qui permet de calculer et afficher une valeur approchée de **GL**. Le calcul s'arrête lorsque la différence entre deux termes consécutifs devient inférieure à **epsilon**.

### Problème : (10 points)

Une **calculatrice**, ou **calculette**, est une machine conçue pour simplifier, et fiabiliser, des opérations de calculs.

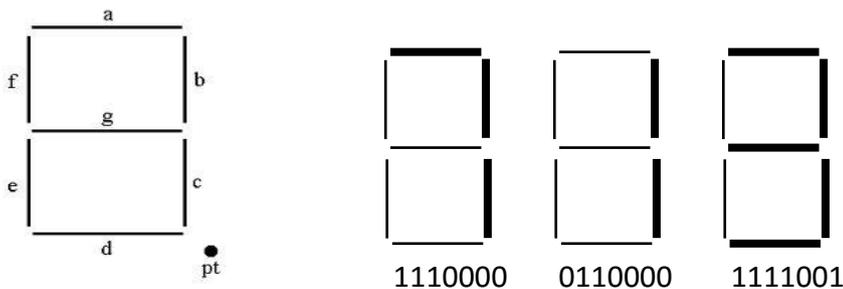
Dans les années 1970, elles se miniaturisent pour devenir portables grâce à l'affichage à **sept segments**

Les afficheurs 7 segments sont un type d'afficheur très présent sur les calculatrices et les montres à affichage numérique : les chiffres s'écrivent en allumant ou en éteignant des segments. Quand les 7 segments sont allumés, on obtient le chiffre **8**.

Un segment allumé est représenté par le caractère **1**

Un segment éteint est représenté par le caractère **0**

Voici quelques exemples représentés avec l'affichage à 7 segments :



- Pour que la calculatrice affiche la valeur 7 il faut que les segments **a,b,c** doivent allumer et les segments **d,e,f,g** éteint (1110000).
- Pour que la calculatrice affiche la valeur 3 il faut que les segments **a,b,c,d,g** doivent allumer et les segments **e , f** éteint.( 1111001)

Etant donnée un fichier d'enregistrement intitulé « **calculatrice.dat** », où chaque enregistrement est composé de deux champs :

- Un champ **chiffre** contenant un chiffre
- Un champ **code** contenant le code de chiffre (suite de 0 et 1) selon l'état des segments allumé ou éteint en commençant par le segment **a** puis **b** ... jusqu'à **g(ordre alphabétique)**

Le fichier « **calculatrice.dat** » est le suivant :

<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>
<b>1111110</b>	<b>0110000</b>	<b>1101101</b>	<b>1111001</b>	<b>0110011</b>	<b>1011011</b>	<b>1011111</b>	<b>1110000</b>	<b>1111111</b>	<b>1110111</b>

En Utilisant le fichier « **calculatrice.dat** », on se propose de calculer les opérations enregistrées dans le fichier texte intitulé « **operations.txt** » et de sauvegarder le résultat de chaque opération dans un fichier texte « **resultat.txt** » .

- chaque ligne du fichier **operations.txt** contient une seule opération.
- chaque chiffre est une chaîne de 7 caractères binaires.

### Exemple :

Si le contenu du fichier « **operation.txt** »

**1101101+011001111110000+1111111111110**  
**1111001+1011111**

Après le calcul et le codage, le programme remplit et affiche le contenu fichier « **resultat.txt** » suivant :

**011000011011011110111**  
**1110111**

En effet :  $\underbrace{1101101}_{2} + \underbrace{011001111110000}_{47} + \underbrace{1111111111110}_{80}$   
 $= 2+47+80= 129$

$\underbrace{0110000}_{1} \quad \underbrace{1101101}_{2} \quad \underbrace{1110111}_{9}$

NB *le candidat n'est pas appelé à remplir le deux fichier « **calculatrice.dat** » et « **operation.txt** »*

### Travail demandé :

1. Analyser le problème en le décomposant en modules
2. Analyser chacun des modules envisagés.

 Bon Travail 