

- ▶ Au cours de ce TP, nous mettons en œuvre quelques idées de base en programmation. L'objectif est de construire un tableau à une dimension, de le remplir de valeurs, puis de calculer un certain nombre de paramètres relatifs à ce tableau : la somme de ses éléments, la valeur du plus grand élément, le nombre d'apparitions de cet élément dans le tableau...

### Les éléments de Maple à connaître

- ▶ Voici d'abord comment définir un tableau  $v$  indexé de 0 à  $N$ , et donner une valeur à certains de ses éléments :
 

```
N := 10 :
v := array(0..N, []) :
v[3] := 1515 : v[7] := 1789 :
```

- ▶ La construction `for ... from ... to ... do ... od` permet de répéter un traitement un nombre déterminé de fois. Par exemple, pour mettre la valeur 0 dans toutes les cases du tableau précédent :

```
for k from 0 to N
do
v[k] := 0
od:
```

**Q1** Définissez dans votre feuille de calcul un tableau  $v$  indexé de 0 à  $N$ , avec  $N = 1000$ . Dans la suite, nous noterons  $v_k$  la valeur contenue dans la case  $v[k]$  du tableau.

**Q2** Consultez la page <http://newbox/div/pcsi2/tpinfo/codes-pcsi2> et relevez le nombre qui est face de votre identifiant ; placez ce nombre dans  $v[0]$ .

**Q3** Définissez dans votre feuille de calcul les valeurs  $a = 239$  et  $m = 673$ . Remplissez ensuite le reste du tableau comme suit : pour  $k \in \llbracket 1, N \rrbracket$ ,  $v_k$  est le reste dans la division de  $av_{k-1}$  par  $m$ . Par exemple, si  $v_3 = 72$ , alors  $v_4 = 383$ . La fonction `irem` permet de calculer ce reste.

**Q4** Calculez la somme des éléments du tableau.

**Q5** Déterminez la plus grande valeur contenue dans le tableau. Vous utiliserez la fonction `max`. Notons  $M$  cette valeur.

- ▶ La construction `if ... then ... else ... fi` permet de choisir l'une des deux branches d'une alternative :

```
if a<b and b<c
then
print "la suite (a,b,c) est croissante"
fi:
```

Ici, nous n'avons précisé que l'action à effectuer lorsque la condition est satisfaite. Voici comment spécifier également une action à effectuer lorsque la condition n'est pas satisfaite :

```
if a<Q
then
compteur_1 := compteur_1 + 1
else
compteur_2 := compteur_2 + 1
fi:
```

Rappel culturel : en français, une *alternative* est un choix entre deux possibilités. Ainsi, dans la chanson *Le gorille*, Brassens dit :

Qu'une alternative pareille un de ces quatre jours m'échoie  
C'est, j'en suis convaincu, la vieille qui sera l'objet de mon choix

**Q6** Déterminez le nombre d'apparitions de  $M$  dans le tableau.

**Q7** Existe-t-il dans le tableau deux éléments consécutifs égaux ?

**Q8** Un *pic* du tableau est un indice  $k \in \llbracket 1, N - 1 \rrbracket$  tel que  $v_{k-1} < v_k$  et  $v_{k+1} < v_k$ . Combien votre tableau comporte-t-il de pics ?

**Q9** Une *montée* du tableau est une suite d'éléments consécutifs  $v_i, v_{i+1}, \dots, v_{j-1}, v_j$  avec  $0 \leq i < j \leq N$ , vérifiant  $v_k < v_{k+1}$  pour  $0 \leq k < N$ . La *longueur* de cette montée est  $j - i + 1$ . Déterminez la longueur de la plus longue montée du tableau.