

Série 10 - Sous-programmes et tri

Exercice 1

On désire développer un programme qui affiche tous les nombres divisibles par 11 dans l'intervalle [A, B] sachant qu'un nombre positif est divisible par 11 si et seulement si, la valeur absolue de la différence entre la somme de ses chiffres de rang impair et la somme de ses chiffres de rang pair est divisible par 11.

Exemples :

919380

$$9 + 9 + 8 = 26 \text{ et } 1 + 3 + 0 = 4$$

$$|26 - 4| = 22 ; 22 \text{ est multiple de } 11$$

$\Rightarrow 919380$ est divisible par 11

2025

$$2 + 4 = 6 \text{ et } 0 + 5 = 5$$

$$|6 - 5| = 11 ; 1 \text{ n'est pas multiple de } 11$$

$\Rightarrow 2025$ n'est pas divisible par 11

1) Soit la fonction **Somme** suivante :

```
Fonction Somme(d: entier, ch: chaîne):entier
Début
  s ← 0
  Pour i de d à long(ch)-1 Faire [pas=2]
    S ← S + Valeur(ch[i])
  Fin Pour
  Retourner s
Fin
```

TDOL

Objet	Type/Nature
i, s	Entier

Remplir le tableau suivant par la valeur de la variable **S** obtenue à la suite de l'appel de la fonction **Somme**.

Appel	Résultat
$S \leftarrow \text{Somme}(0, "241863")$	
$S \leftarrow \text{Somme}(1, "241863")$	

2) Récrire la fonction **Somme** tout en remplaçant la boucle **Pour** par une structure itérative à condition d'arrêt.

```
Fonction Somme(d: entier, ch: chaîne):entier
```

Début

$s \leftarrow 0$

.....
.....
.....
.....
.....

```
    Retourner S
```

```
Fin
```

- 3) La fonction **Calcul** suivante permet de vérifier si un entier **N** positif est divisible par **11** ou non

```
Fonction Calcul(N: entier):..... // (1) type de retour
```

Début

```
    ch < ..... // (2) Convertir n en chaîne
    s1 < ..... // (3) Somme des chiffres de ch, de rang pair
    s2 < ..... // (4) Somme des chiffres de ch, de rang impair
    diff < abs(s1 - s2)
    Retourner ..... // (5) retourner le résultat
```

Fin

Encercler la bonne réponse afin de compléter la fonction **Calcul**.

N° Instruction	Réponse 1	Réponse 2	Réponse 3
1	Entier	Booléen	Chaine
2	Convch(N)	Estnum(N)	Valeur(N)
3	Somme(0, ch)	Somme(1, ch)	Somme(N div 2, ch)
4	Somme(0, ch)	Somme(1, ch)	Somme(N div 2, ch)
5	diff mod 11=0	diff div 11 = 0	diff

- 4) L'algorithme du programme principal affiche tous les nombres divisibles par 11 dans l'intervalle **[A, B]** en exploitant la fonction **Calcul**.

```
Algorithme Divisible11
```

Début

```
// Séquence 1
Lire(A)
Lire(B)

// Séquence 2
Si A>B alors
    Permut(A,B)
Fin Si

// Séquence 3
ch < 0
Pour i de 0 à B faire
    Si Calcul(A, B) = 0 alors
        ch < ch + i + "|"
    Fin Si
Fin Pour
Ecrire ("divisibles par 11")
Fin
```

En se référant à l'algorithme **Divisible11** et pour chacune des propositions suivantes, barrer la réponse incorrecte :

Proposition		
	Formels	Effectifs
Les paramètres d'appel de la procédure Permut sont des paramètres :		
Le mode de passage des paramètres dans l'entête de la procédure Permut est par :	Valeur	Adresse
Ch est une variable :	Globale	Locale

- a. Dresser le T.D.O.G de l'algorithme Divisible11.
- b. Transformer la **séquence 1** en une procédure en ajoutant les contrôles nécessaires, sachant que **A** et **B** sont deux entiers strictement positifs et distincts.
- c. Développer la procédure **Permut** qui permet d'échanger les valeurs des entiers **A** et **B**.
- d. Corriger la **séquence 3** dont le rôle est d'afficher les entiers de l'intervalle **[A, B]** divisibles par 11.

Exercice 2 – Ligue des champions

La première division est composée de **N** clubs sélectionnés à la fin de chaque saison. Le **club champion** et **vice-champion** sont qualifiés pour **la Ligue des champions de la CAF** et le **troisième** pour **la coupe de la CAF** tandis que les **deux derniers du classement** sont relégués en **division 2**.

On désire développer l'algorithme d'un programme, simulant le championnat de Tunisie de football, qui :

- Remplit un tableau **TC** par les **abréviations des noms des clubs**, avec (**4 ≤ N ≤ 16**). Le **nom d'un club** est **non vide**, composée uniquement par des **lettres alphabétiques majuscules** et **différent des autres**.
NB : Le candidat est appelé à utiliser la fonction **alpha(ch)** sans la développer. Cette fonction booléenne retourne **Vrai** si la chaîne est composée par des lettres majuscules, faux sinon.
- Remplit un tableau **TM** par les **résultats des matchs** de chaque équipe (**aller/retour**). Le résultat d'un match est :
 - "0" pour un **match perdu**.
 - "1" pour un **match gagné**.
 - "X" pour un **match nul**.

Chaque équipe joue **(N-1) * 2** matchs, contre les autres équipes.

- Calcule les **scores des équipes** dans le tableau **TS** en fonction de leurs résultats durant les **(N-1) * 2** matchs joués. L'équipe obtient :
 - **0** points si elle **perd un match**.
 - **3** points si elle **gagne un match**.
 - **1** point si elle fait un **match nul**.

Exemple : pour les résultats "**10XX10**" le score sera : **3 + 0 + 1 + 1 + 3 + 0 = 8**

- Affiche :
 - Les **deux équipes qualifiées** pour la **Ligue des champions de la CAF** (**1^e et 2^e places**).
 - L'**équipe qualifiée** pour la **coupe de la CAF** (**3^e place**).
 - Les **deux équipes reléguées** en **Ligue II** (**les 2 dernières du classement**)

Exemple : pour **N=6** et les 3 tableaux **TC**, **TM** et **TS** suivants :

TC	OB	ESS	JSK	CA	CSS	EST
	0	1	2	3	4	5
TM	011X0XX0X0	1111X11111	00100X0000	001XX101X0	X01XXX01X0	1X1XX101101
	0	1	2	3	4	5
TS	10	28	4	12	11	21
	0	1	2	3	4	5

Le programme affichera :

Clubs qualifiés pour la Ligue des champions de la CAF : ESS et EST

Club qualifié pour la coupe de la CAF : CA

Clubs relégués en Ligue II : OB et JSK

Travail demandé :

1. Ecrire l'algorithme du programme principal, solution à ce problème, en le décomposant en modules.
2. Ecrire l'algorithme de chacun des modules envisagés.