

COMMISSARIAT RÉGIONAL DE L'ÉDUCATION – KÉBILI □ □ □ □ □ □ DEVOIR DE SYNTHÈSE N° 2 ÉPREUVE : Informatique	Année Scolaire : 2025/2026	
	Sections : Sciences Expérimentales, Mathématiques et Sciences Techniques	
	Niveau : 4^{ème} Année	Durée : 1 h 30 mn
	Date : Jeudi, 05/03/2026 – 8h:15mn	

Nom et prénom : – Classe :

Le sujet comporte cinq pages numérotées de 1/5 à 5/5

Les réponses à l'exercice 1, l'exercice 2 et l'exercice 3 doivent être rédigées sur cette même feuille qui doit être remise avec la copie d'examen à la fin de l'épreuve.

EXERCICE 1: (4 POINTS)

1) Pour chacun des séquences ci-dessous, mettre dans chaque case la lettre **V** si la proposition (rôle de la séquence) est correcte ou la lettre **F** dans le cas contraire :

Séquence 1 :

Ch1 □ ""

Pour i de 0 à Long (ch) - 1 **Faire**

Si "a" ≤ ch [i] ≤ "z" **Alors**

ch1 □ ch1+Chr (Ord (ch[i]) - 32)

Sinon

ch1 □ ch1 + ch [i]

FinSi

Fin Pour

Retourner le code ASCII d'un caractère

Convertir la chaine ch en majuscule

Convertir la chaine ch en minuscule

Séquence 2 :

x □ ""

i □ a

Tant que (i ≥ a) et (i ≤ b-1) **Faire**

x □ x + ch [i]

i □ i+1

Fin Tant que

Insérer la chaine x dans la chaine ch à la position a

Effacer une partie de la chaine ch à partir de position a jusqu'à b-1

Retourner une partie de la chaine ch à partir de position a jusqu'à b-1

Séquence 3 :

```

R  $\square$  Vrai
i  $\square$  0
Tant que R = Vrai Et i < Long (ch) Faire
    Si Non ( ch[i]  $\in$  [ "0" , "9" ] ) Alors
        R  $\square$  Faux
    FinSi
    i  $\square$  i + 1
Fin Tant que

```

	Vérifier si la chaine contient seulement des chiffres
	Vérifier si la chaine contient seulement des lettres
	Vérifier si la chaine ne contient pas de chiffres

Séquence 4 :

```

x1  $\square$  ""
x2  $\square$  ""
i  $\square$  a
Tant que i < a Faire
    x1  $\square$  x1 + ch[ i ]
    i  $\square$  i+1
Fin Tant que
i  $\square$  b
Tant que i < Long (ch) Faire
    x2  $\square$  x2 + ch [ i ]
    i  $\square$  i + 1
Fin Tant que
ch  $\square$  x1 + x2

```

	Concaténer deux chaines
	Effacer une partie de la chaine ch à partir du caractère d'indice a jusqu'à le caractère d'indice b-1
	Copier une partie de la chaine ch à partir du caractère d'indice a jusqu'à le caractère d'indice b-1

2) Remplacer chaque séquence par l'appel d'une fonction prédéfinie :

- Séquence 1 : ch1 $\square$
- Séquence 2 : x $\square$
- Séquence 3 : R $\square$
- Séquence 4 : ch $\square$

Exercice 2 : (5 POINTS)

Soit l'algorithme du programme principal suivant :

```

ALGORITHME Traitement
DEBUT
    Écrire ( " saisir x : " )
    Lire ( x )
    Si Inconnue ( x ) Alors
        Écrire ( x , " vérifie la propriété " )
    Sinon
        Écrire ( x , " ne vérifie pas la propriété " )
    FinSi
FIN

```

Soit l'algorithme suivant de la fonction **Inconnue** :

```

Fonction Inconnue ( y : ..... ) : .....
..
DEBUT
  Tant que ( Long (y) > 0 ) Et ( "0" ≤ y [ 0 ] ≤ "9" ) Faire
    y □ Effacer ( y , 0 , 1 )
  Fin Tant que
  R □ Long ( y ) = 0
  Retourner R
FIN

```

1) Compléter l'entête de la fonction Inconnue.

Fonction Inconnue (y :) :

2) Tracer le tableau de déclaration des objets globaux **T.D.O.G** du programme principal **Traitement**.

Objet	Nature/type

3) Tracer le tableau de déclaration des objets locaux **T.D.O.L** relatif à la fonction **Inconnue**.

Objet	Nature/type

4) A partir des algorithmes ci-dessus, remplir le tableau suivant :

Paramètres formels	
Paramètres effectifs	
Fonctions prédéfinies	
Expressions booléennes	

5) Donner le résultat affiché par le programme **Traitement** pour chacune des valeurs suivantes :

x	Résultat
"-1450"	
"1450"	

"14.50"	
"14+50"	

6)Déduire le rôle de la fonction **Inconnue** :

.....

.....

EXERCICE 3 : (3POINTS)

Soit l'algorithme de la procédure **Quoi** suivant :

```

Procédure Quoi ( @ R : Chaîne ; N , B : Entier )
DEBUT
  Code ← "0123456789ABCDEFGH"
  R ← ""
  Répéter
    R ← Code [ N mod B ] + R
    N ← N Div B
  Jusqu'a ( N = 0 )
FIN

```

1. Montrer pour les cas suivants le résultat renvoyé par la procédure **Quoi** en complétant le tableau de la trace d'exécution.

1^{er} cas : N = 100 , B = 16 : Quoi (R , N , B) → résultat : R = "64"

R						
N						

2^{ème} cas : N = 75 , B = 8 : Quoi (R , N , B) → résultat : R = "113"

R						
N						

3^{ème} cas : N = 45 , B = 2 : Quoi (R , N , B) → résultat : R = "101101"

R	"1"	"01"	"101"	"1101"	"01101"	"101101"
N	22	11	5	2	1	0

4^{ème} cas : N = 150 , B = 7 : Quoi (R , N , B) → résultat : R = "303"

R						
N						

5^{ème} cas : N = 44 , B = 15 : Quoi (R , N , B) → résultat : R = "2E"

R						
N						

2. Dédurre le rôle de la procédure **Quoi** :

.....

EXERCICE 4 : (8 POINTS)

Lors d'une compétition d'une course de **10 km** en pied, on souhaite gérer les performances des participants. Dès que le joueur termine la course, on saisit sa durée réalisée sous la forme "**mm:ss**".

Dans ce cadre on se propose d'écrire un algorithme d'un programme intitulé **Course** qui permet de :

- Saisir le nombre de participants **N** ($5 \leq N \leq 100$).
- Remplir un tableau **TJ** par les noms des participants (**le nom ne contient que des lettres et des espaces**).
- Remplir un tableau **TH** par les durées réalisées, on suppose qu'il existe une fonction **Verif** qui permet de vérifier la validité des éléments de **TH**.
 (**Fonction Verif (ch : Chaîne) : Booléen**)
- Afficher le résultat de la compétition en indiquant le vainqueur de la compétition (Nom, prénom et le temps réalisé), pour les autres participants, l'ordre, temps réalisé et l'écart du temps avec le vainqueur (gagnant).

Exemple : Pour N=6

TJ :

TJ [0]	TJ [1]	TJ [2]	TJ [3]	TJ [4]	TJ [5]
Taieb Mohamed	Cherif Malek	Trabelsi Yassine	Bakouri Khaled	Dhaouadi Amine	Ktata Marwen

TH :

TH [0]	TH [1]	TH [2]	TH [3]	TH [4]	TH [5]
"27:45"	"30:08"	"29:35"	"27:14"	"28:30"	"30:53"

Le programme affiche :

- 1- Vainqueur : Bakouri Khaled 27:14
- 2- Taieb Mohamed (27:45 ; 00:31)
- 3- Dhaouadi Amine (28:30 ; 01:16)
- 4- Trabelsi Yassine (29:35 ; 02:21)
- 5- Cherif Malek (30:08 ; 02:54)

6- Ktata Marwen (30:53 ; 03:39)

Travail demandé :

1. Ecrire l'algorithme du programme principal en le décomposant en modules.
2. Ecrire les algorithmes des modules envisagés.

NB : *le candidat n'est pas appelé à développer l'algorithme de la fonction Verif*