

Commissariat Régional de l'Éducation de Sousse		
DEVOIR DE SYNTHÈSE N°3	Sections : Mathématiques, Sciences Expérimentales et Sciences Techniques	
	Épreuve : Informatique	
	Date : Mardi, 14 mai 2024	Durée : 1h 30mn
Nom et Prénom : Classe :		Note :/20

Le sujet comporte 4 pages numérotées de 1/4 à 4/4.

Les réponses à l'exercice n°1 et à l'exercice n°2 doivent être rédigées sur cette même feuille qui doit être remise avec la copie du devoir à la fin de l'épreuve.

❧❧❧❧★❧❧❧❧

EXERCICE N°1 (2.5 points)

Le jeu **ABRA-CADABRA** consiste à repérer le gobelet où se cache la balle après un certain nombre de permutations. Pour simuler ce jeu, on propose l'algorithme ci-dessous du programme principal nommé **JEUX** qui utilise deux tableaux nommés **Goblet** et **Couleur**, et fait appel aux deux procédures **ABRA** et **CADABRA**.

Algorithme JEUX

Début

```

Goblet [0] ← Vrai
Couleur [0] ← "Rouge"
Goblet [1] ← Faux
Couleur [1] ← "Vert"
Goblet [2] ← Faux
Couleur [2] ← "Bleu"
Lire (N)
Pour i de 1 à N faire
    ABRA (Goblet [0], Goblet [1])
    CADABRA (Goblet [1], Goblet [2])
    ABRA (Goblet [2], Goblet [0])
Fin Pour
SI Goblet [0] ALORS
    Écrire ("La balle est sous le gobelet ", Couleur [0])
SINON SI Goblet [1] ALORS
    Écrire ("La balle est sous le gobelet ", Couleur [1])
SINON SI Goblet [2] ALORS
    Écrire ("La balle est sous le gobelet ", Couleur [2])
FIN SI

```

Fin



Procédure ABRA (@ X, Y : booléen)

Début

```

AUX ← X
X ← Y
Y ← AUX

```

Fin

Procédure CADABRA (X, Y : booléen)

Début

```

AUX ← X
X ← Y
Y ← AUX

```

Fin

Pour chacune des propositions ci-dessous, **une seule réponse est correcte**. Mettre une croix (X) dans la case correspondante à la bonne réponse.

1) Les objets X et Y utilisés dans les entêtes des procédures **ABRA** et **CADABRA** sont des :

- | | |
|--|----------------------|
| | variables globales |
| | paramètres formels |
| | paramètres effectifs |

2) Le mode de passage des paramètres utilisé dans la procédure **CADABRA** est :

- ☐ par valeur
- ☐ par référence
- ☐ par adresse

3) Pour **N=1**, et après exécution du programme **JEUX**, le message affiché sera :

- ☐ La balle est sous le gobelet Rouge
- ☐ La balle est sous le gobelet Vert
- ☐ La balle est sous le gobelet Bleu

4) Pour **N=2**, et après exécution du programme **JEUX**, le message affiché sera :

- ☐ La balle est sous le gobelet Rouge
- ☐ La balle est sous le gobelet Vert
- ☐ La balle est sous le gobelet Bleu

5) Pour générer aléatoirement une valeur de **N** comprise entre **1** et **5**, l’instruction **Lire(N)** doit être remplacée par l’instruction :

- ☐ **N ← Aléa (1,5)**
- ☐ **N ← Aléa (1,6)**
- ☐ **N ← Aléa (0,5)**

EXERCICE N°2 (4.5 points)

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

```

Fonction Inconnue .....
Début
    Verif ← Vrai
    Pour i de 0 à long(X)-1 Faire
        Si Non(X[i] dans ["0".."9"]) Alors
            Verif ← Faux
        Fin Si
    Fin Pour
    Retourner Verif
Fin
  
```

- 1) Compléter l’entête de la fonction **Inconnue**.
- 2) Compléter le **tableau de déclaration des objets locaux** de cette fonction.

T.D.O.L

<i>Objet</i>	<i>Type / Nature</i>

3) Remplir le tableau ci-dessous par le résultat retourné par la fonction **Inconnue** pour chacune des valeurs de **X** proposées.

Valeur de X	Résultat retourné
"125,45"	
"123"	
"12A3"	

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

- Page 3 sur 4 -

Exemples :

- Le drone "**A9:50#2:30**" a pour identifiant "**A9**", le poids maximal qu'il peut transporter est **50** kg et lui sont affectés, à cet instant, **2** colis ayant un poids total de **30** Kg.
- Le drone "**Z2:90#0:0**" a pour identifiant "**Z2**", le poids maximal qu'il peut transporter est **90** kg et aucun colis ne lui est affecté à cet instant.

Lors de l'affectation d'un nouveau colis à l'un des drones, les contraintes suivantes doivent être respectées :

- La somme des poids des colis à livrer (**PoidsColis**) ne doit pas dépasser le poids maximal (**PoidsMax**) que peut transporter un drone.
- Le nombre des colis à transporter par un drone (**NbrColis**) doit être inférieur ou égal à **9**.
- Un nouveau colis à livrer est affecté au premier drone qui peut le transporter.

On se propose d'écrire un programme qui permet de remplir le tableau **DL** par **N** drones ($3 < N \leq 20$), d'affecter un nouveau colis à livrer de poids **P** ($0 < P \leq 90$) à un drone et d'afficher :

- le message "**Colis affecté avec succès, le nouvel état du tableau est :**", suivi du nouvel état du tableau **DL** après avoir mis à jour le nombre de colis transportés et la somme de leurs poids, si l'affectation est possible,
- le message "**Impossible de livrer le colis pour l'instant**", dans le cas contraire.

N.B. : Les identifiants des drones (**IdDrone**) doivent être distincts.

Exemple : Pour **N=4** et le tableau **DL** suivant à un instant donné :

DL	Z2:70#4:65	P7:90#9:70	A9:80#2:20	L3:70#0:0
	0	1	2	3

- Pour un nouveau colis à livrer de poids **P** égal à **15** Kg, le programme affiche :

Colis affecté avec succès, le nouvel état du tableau est :

Z2:70#4:65
P7:90#9:70
A9:80#3:35
L3:70#0:0

En effet, le troisième drone peut transporter le nouveau colis puisqu'il satisfait toutes les contraintes.

- Pour un nouveau colis à livrer de poids **P** égal à **80** Kg, le programme affiche :

"Impossible de livrer le colis pour l'instant"

En effet, aucun drone, à cet instant, ne peut transporter le colis car la somme des poids des colis à livrer (**PoidsColis**) de chacun dépasse son poids maximal (**PoidsMax**) qu'il peut transporter.

Travail demandé :

- 1) Écrire un algorithme du programme principal, solution à ce problème, en le décomposant en modules.
- 2) Écrire un algorithme pour chaque module envisagé.

N.B. : Le candidat est appelé à dresser les tableaux de déclaration des objets et des nouveaux types nécessaires.