

Section : ..... N° d'inscription : ..... Série : .....

Nom et prénom : .....

Date et lieu de naissance : .....

Signature des  
surveillants

.....  
.....

Épreuve : Algorithmique et Programmation - Section : Sciences de l'informatique - Session de contrôle 2021

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

Les pages 1/4 et 2/4 sont à remplir par le candidat et à rendre avec sa copie

### Exercice 1 : (4,5 points)

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

```
0) DEF PROC Traitement (Struct1 : X ; n, m : Entier ; Var Struct2 : X)
1) Pour i de 1 à n Faire
    Pour j de 1 à m Faire
        Struct2[j,i] ← Struct1[i,j]
    Fin Pour
Fin Pour
2) Fin Traitement
```

**N.B. :** **n** et **m** sont deux entiers de l'intervalle [1,10].

1. En se basant sur l'algorithme du module **Traitement** et pour chacune des questions suivantes, valider chaque réponse par la lettre "V" si elle est correcte ou par la lettre "F" dans le cas contraire.

Question	Réponse
a) Quelle est la nature de X ?	<input type="checkbox"/> Type de données <input type="checkbox"/> Variable <input type="checkbox"/> Constante
b) Quelle est la déclaration valide de X ?	<input type="checkbox"/> Tableau [1..10, 1..10] d'entiers <input type="checkbox"/> Tableau [1..10] d'entiers <input type="checkbox"/> Entier
c) Quel sera le type de données le plus adéquat de <b>n</b> et <b>m</b> afin d'optimiser l'espace mémoire utilisé ?	<input type="checkbox"/> Entier <input type="checkbox"/> Octet <input type="checkbox"/> Entier long
d) Dans un programme appelant le module <b>Traitement</b> a été appelé comme suit : <b>Traitement (S1, p, q, S2)</b> Qu'appelle-t-on l'objet <b>p</b> ?	<input type="checkbox"/> Paramètre effectif <input type="checkbox"/> Paramètre formel <input type="checkbox"/> Variable locale du module Traitement
e) Soit l'appel suivant : <b>Traitement (S2, p, q, S1)</b> Quel est l'objet qui contient le résultat obtenu par l'appel du module ?	<input type="checkbox"/> S1 <input type="checkbox"/> S2 <input type="checkbox"/> p



Ne rien écrire ici

2. Quel est le rôle de la procédure **Traitement** ?

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

**Exercice 2 : (3 points)**

Afin d'automatiser la gestion des rendez-vous d'un personnel administratif, on se propose d'utiliser un tableau **T** contenant les informations de **N** ( $2 \leq N \leq 50$ ) rendez-vous. Chaque rendez-vous est un enregistrement formé de trois champs : le numéro du jour **NJ**, le numéro du mois **NM** et la description **Desc** du rendez-vous.

**Travail demandé :**

1. Compléter le tableau de déclaration des nouveaux types suivant par une déclaration d'un type pour le tableau **T** ainsi que les types nécessaires à sa déclaration.

Types	
.....	.....
.....	.....
.....	.....
.....	.....
.....	.....
.....	.....

2. Ecrire un algorithme d'une procédure **Tri\_RDV(T, N)** qui permet de trier les rendez-vous contenus dans le tableau **T** dans l'ordre croissant du numéro du mois **NM** et en cas d'égalité le tri sera effectué dans l'ordre croissant du numéro du jour **NJ**.

**N.B. :** Le candidat n'est pas appelé à saisir **T** et **N**.

**Exemple :** Pour **N = 5** et le tableau **T** suivant :

1	2	3	4	5
NJ 12	NJ 25	NJ 4	NJ 12	NJ 10
NM 5	NM 3	NM 5	NM 3	NM 5
Desc Atelier	Desc Réunion	Desc Entretien	Desc Réunion	Desc Atelier

Le tableau **T** sera trié comme suit :

NJ 12	NJ 25	NJ 4	NJ 10	NJ 12
NM 3	NM 3	NM 5	NM 5	NM 5
Desc Réunion	Desc Réunion	Desc Entretien	Desc Atelier	Desc Atelier



<b>RÉPUBLIQUE TUNISIENNE</b>  <b>MINISTÈRE DE L'ÉDUCATION</b>	<b>EXAMEN DU BACCALAURÉAT</b> <b>SESSION 2021</b>	<b>Session de contrôle</b>
	Épreuve : <b>Algorithmique et Programmation</b>	Section : <b>Sciences de l'informatique</b>
	Durée : <b>3h</b>	Coefficient de l'épreuve : <b>2.25</b>

N° d'inscription



**Important :**

Chaque solution développée par le candidat sous forme d'un algorithme doit être accompagnée d'un tableau de déclaration des objets ayant la forme suivante :

Objet	Type/Nature	Rôle

**Exercice 3 : (5,5 points)**

Un entier naturel non nul **Nbre** est dit **nombre triangulaire**, s'il existe un entier naturel **n** tel que :

$$\text{Nbre} = 1 + 2 + 3 + \dots + n = n * (n+1) / 2$$

Un entier naturel non nul **Nbre** est dit **nombre triangulaire carré** lorsqu'il est un nombre triangulaire et il existe un entier **m** tel que  $\text{Nbre} = m^2$ .

**Exemples :**

- 5 n'est pas un nombre triangulaire car  $1 + 2 = 3$  et  $1+2+3 = 6$ , donc il n'existe pas un entier **n** tel que  $5 = n * (n+1)/2$
- 6 est un nombre triangulaire car  $6 = 1 + 2 + 3 = (3*4)/2$  mais il n'est pas le carré d'un autre entier, donc 6 n'est pas un nombre triangulaire carré.
- 36 est un nombre triangulaire car  $36 = 1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 = (8*9)/2$  et il est égal au carré de 6 ( $36 = 6^2$ ), donc 36 est un nombre triangulaire carré.

**Travail demandé**

1. Ecrire un algorithme d'une fonction **Triangulaire (Nbre)** qui permet de retourner la valeur de l'entier **n** si un entier **Nbre** est un nombre triangulaire et **0** dans le cas contraire.
2. En utilisant la fonction **Triangulaire**, écrire un algorithme d'une fonction **Triangulaire\_carré (Nbre)** qui permet de retourner l'entier **m** si un entier **Nbre** est un nombre triangulaire carré et **0** dans le cas contraire.
3. En faisant appel aux deux fonctions précédentes, écrire un algorithme d'une procédure **Remplir (T, k)** qui permet de remplir un tableau **T** d'enregistrements par les **k** premiers nombres triangulaires carrés où chaque enregistrement est formé de trois champs contenant respectivement le nombre **Nbre**, l'entier **n** et l'entier **m**.

**N.B. :**

- Le candidat n'est pas appelé à saisir **k**.
- **T** est de type **Tab**.

**Exercice 4 : (7 points)**

Parmi les méthodes qui ont été développées pour trouver les termes de la suite de **Thue Morse**, on cite les deux méthodes suivantes :



### 1<sup>ère</sup> méthode :

On suppose que le premier terme de la suite est  $U_0 = 0$ .

Pour calculer le terme  $U_n$  il suffit de concaténer les  $nb$  bits  $t_0, t_1, \dots, t_{nb-1}$  avec  $nb = 2^n$ . On obtient ainsi :

$U_n = t_0 t_1 \dots t_{nb-1}$ , sachant que :

- $t_0 = 0$
- $t_i = \begin{cases} t_{(i \text{ div } 2)} & \text{si } i \text{ est pair} \\ \text{ou} \\ 1 - t_{(i \text{ div } 2)} & \text{si } i \text{ est impair} \end{cases}$

### Exemple :

Pour le terme  $U_3$  le programme affichera  $U_3 = 01101001$

En effet :

- Le nombre de bits  $nb$  de  $U_3$  est 8 car  $nb = 2^n = 2^3 = 8$
- Le terme  $U_3$  est obtenu par la concaténation des 8 premiers bits  $t_0, t_1, \dots, t_7$ , avec :  
 $t_0 = 0, t_1 = 1 - t_0 = 1, t_2 = t_1 = 1, t_3 = 1 - t_1 = 0, t_4 = t_2 = 1, t_5 = 1 - t_2 = 0, t_6 = t_3 = 0, t_7 = 1 - t_3 = 1$

### 2<sup>ème</sup> méthode :

On suppose que le premier terme de la suite est  $U_0 = 0$ .

Le nombre de bits du terme  $U_n$  de la suite est égal à  $2^n$ . Pour le calculer on procède comme suit :

- Pour chaque indice  $i$  allant de 0 à  $2^n - 1$  :
  - ✓ Convertir  $i$  en binaire.
  - ✓ Déterminer la somme  $S$  des chiffres de l'équivalent binaire de  $i$ .
  - ✓ Le bit d'indice  $i$  reçoit le reste de la division entière de  $S$  par 2.
- Le terme  $U_n$  de la suite est la concaténation des restes obtenus.

### Exemple :

Pour le terme  $U_3$  le nombre de bits est  $2^3 = 8$ .

$i =$ Indice	0	1	2	3	4	5	6	7
$ib =$ Indice en binaire	0	1	10	11	100	101	110	111
$S =$ Somme des chiffres de $ib$	0	1	1	2	1	2	2	3
$S \bmod 2$	0	1	1	0	1	0	0	1

Le terme  $U_3$  de la suite est alors  $U_3 = 01101001$

### Travail demandé :

1. Ecrire un algorithme d'une fonction intitulée **Methode1** qui permet de déterminer le terme  $U_n$  de la suite de Thue Morse en utilisant le principe de la première méthode décrit précédemment avec  $n$  est un entier positif saisi dans le programme appelant.
2. Ecrire un algorithme d'une fonction intitulée **Methode2** qui permet de déterminer le terme  $U_n$  de la suite de Thue Morse en utilisant le principe de la deuxième méthode décrit précédemment avec  $n$  est un entier positif saisi dans le programme appelant.
3. En faisant appel à l'une des deux fonctions précédentes, écrire un algorithme d'une procédure intitulée **Remplir** qui permet de remplir un fichier texte  $F\_T$  par les  $k$  ( $2 \leq k \leq 10$ ) premiers termes de la suite de Thue Morse à raison d'un terme par ligne.

**N.B. :** Prévoir la saisie de  $k$  dans le module **Remplir**.

### Exemple :

Pour  $k = 4$ , le contenu du fichier  $F\_T$  sera :

```
0
01
0110
01101001
```

