

<b>Devoir de Synthèse N°3</b>		
<b>Commissariat Régional de Bizerte</b> *** <b>Lycée Menzel Bourguiba</b>	<i>Section</i> : Sciences de l'informatique	
	<i>Matière</i> : Algorithmique et programmation	
<i>Niveau</i> : 4 <sup>ème</sup> Année	<i>Date</i> : 05 Mai 2016	<i>Durée</i> : 3Heures

### Exercice 1 :(2,5 points)

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

**0) DEF FN Inconnue ( n,e :entier) :.....**

**1) Si n=0 alors**

Inconnue←""

**Sinon**

r←n mod e

**si r>9 alors**

c←Chr(r+55)

**sinon**

Convch(R,c)

**Fin Si**

Inconnue← FN Inconnue (n div e, e)+c

**Fin Si**

**2) Fin Inconnue**

### Travail demandé :

1. Déterminer le type de retour de la fonction **Inconnue**.
2. S'agit-il d'un procédé itératif ou récursif ? Justifier votre réponse.
3. Donner le résultat retourné par la fonction Inconnue pour chacun des couples (n,e) suivants :**(10,2)** et **(183,16)**
4. En déduire le rôle de cette fonction.

### Exercice 2 :(2,5 points)

Soit la suite suivante définie par :

$$\begin{cases} U_0 = 1 \\ U_n = \frac{(-2)^n}{2 * U_{n-1}} \end{cases}$$

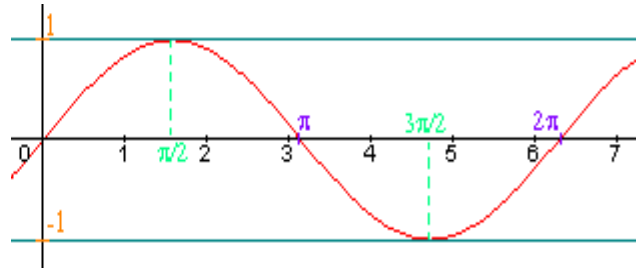
### Travail demandé :

1. Donner la valeur de  $U_1$  et de  $U_2$ .
2. Quel est l'ordre de récurrence de cette suite .justifier votre réponse.
3. Ecrire une analyse d'un module permettant de déterminer pour un entier n le terme  $U_n$

### Exercice 3:(5 points)

Il existe une méthode pour calculer une valeur approchée de  $\pi$  en utilisant la fonction **Sin(x)**. Cette méthode utilise le fait que  $\pi$  est une solution de la fonction **Sin(x)**. De plus, cette solution se trouve entre 2 et 4.

La fonction Sin(x) est strictement décroissante dans l'intervalle **[2..4]**. On vous donne la représentation graphique de la fonction Sin(x) dans cet intervalle. Comme  $\pi$  se trouve dans l'intervalle **[2..4]**, on désire écrire un programme qui permet de chercher la valeur approchée de  $\pi$  en utilisant la méthode suivante :



- Diviser l'intervalle **[2..4]** en **n** sous intervalles avec n entier donné compris entre 10 et 100
- Sauvegarder tous les abscisses (  $x_i$  )  $1 \leq i \leq n$  dans un vecteur de n d'enregistrements au 1<sup>er</sup> champ, l'abscisse x commence par 2.
- Déterminer l'image de chaque abscisse par la fonction Sin(x) et la sauvegarder dans le 2<sup>ème</sup> champ.
- Incrémenter x par le pas **p** avec p=largeur du sous intervalle.
- Finalement la valeur approchée de  $\pi$  est égale à l'abscisse qui correspond à la valeur minimale de la valeur absolue de son image dans le vecteur d'enregistrement

### Travail demandé :

- Ecrire un algorithme d'une procédure **Remplir\_tab(n,T)** permettant de remplir le vecteur d'enregistrements par les n abscisses et leurs images.
- Ecrire un algorithme d'une fonction **Calcul\_pi(n,T)** qui permet de retourner une valeur approchée de  $\pi$  .

### Exercice 4:(10 points)

Mounir a toujours des problèmes, et il veut les résoudre à l'aide de son ordinateur .Il envoie tout le temps des messages à son ami Ahmed, mais vu que les messages transmis contiennent des informations importantes, ils ont peur du piratage. Pour cela, Mounir et son ami se sont mis d'accord à coder leurs messages de la façon suivante :

- **Etape 1** : saisir un nombre M compris entre 10 et 15
  - **Etape2** :
    - a- Calculer NBC=Nombre de Caractères dans le message et NBL=nombre de lignes dans le message.
    - b- Calculer la valeur  $N = (NBC + NBL - 1) \text{ DIV } M$
    - c- Si le reste de la division précédente est différent de zéro, ajouter 1 à N.
  - **Etape 3** : Remplir une matrice de taille NxM par les caractères du message à envoyer en parcourant la matrice ligne par ligne, en remplaçant chaque retour à la ligne par le caractère « # » et en complétant la dernière ligne par des étoiles « \* » si nécessaire.
  - **Etape 4** : Construire un fichier texte structuré de la façon suivante :
    - La 1<sup>ère</sup> ligne contiendra le code hexadécimal du nombre M saisi au début.
    - Chacune des lignes suivantes contiendra une chaîne formée par la concaténation des caractères d'une colonne de la matrice (dans l'ordre).
- ☞ Le contenu de ce fichier texte est le message à envoyer.

**Exemple** : Soit le fichier texte contenant le message à coder suivant :

Bonjour Ahmed,  
 Compte = Mounir123  
 Mot de passe = m80a531

Si M=10

Le nombre de caractères dans ce message est NBC=53 et le nombre de lignes est NBL=3

$N = (NBC + NBL - 1) \text{ DIV } 10 = (53 + 3 - 1) \text{ DIV } 10 = 5$  et le reste = 6

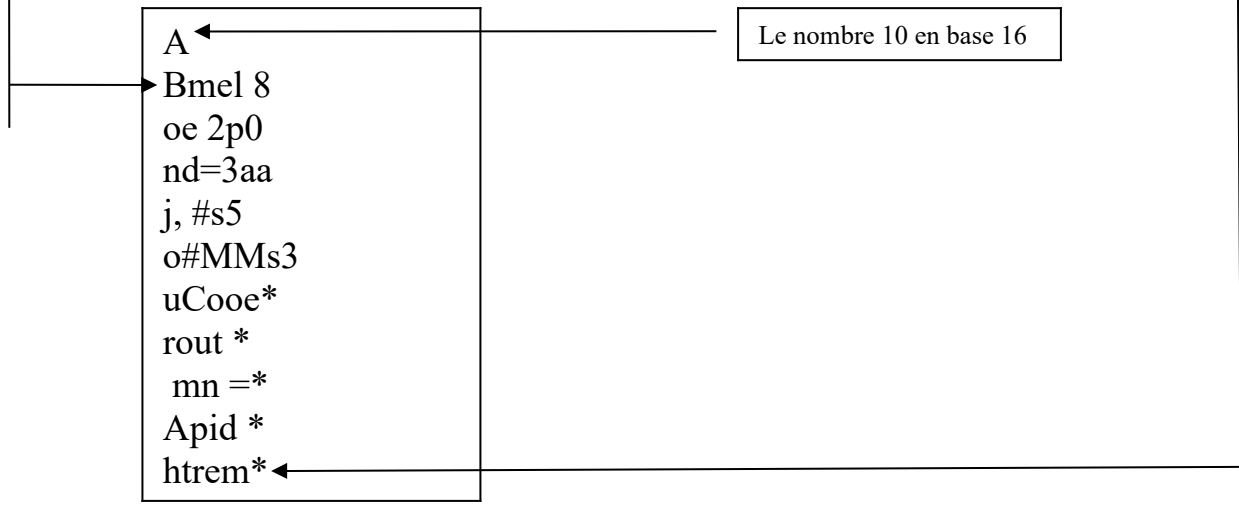
$N = 5 + 1 = 6$

La matrice à construire sera de taille 6x10 :

Le retour à la ligne sera remplacé par le caractère « # » et la dernière ligne sera complétée par des étoiles « \* »

B	o	n	j	o	u	r		A	h
m	e	d	,	#	C	o	m	p	t
e		=		M	o	u	n	i	r
1	2	3	#	M	o	t		d	e
	p	a	s	s	e		=		m
	0	a	5	3	*	*	*	*	*

Le fichier résultat contiendra alors :



Le but de Mounir consiste à réaliser un programme permettant de construire, à partir du fichier texte « C:\M1.txt » contenant déjà le message à envoyer, un deuxième fichier « C:\M2.txt » contenant le message codé obtenu suite à l'application du principe décrit ci-dessus.

### Travail demandé :

- 1) Analyser ce problème en le décomposant modules.
- 2) Analyser chacun des modules envisagés précédemment.