

EXAMEN DU BACCALAUREAT - SESSION DE JUIN 2010

SECTION : Sciences de l'informatique

EPREUVE : Algorithmique et programmation **DUREE :** 3h **COEFFICIENT :** 2,25

Le sujet comporte 3 pages numérotées de 1/3 à 3/3

(Toutes les réponses se feront sur votre feuille de copie)

Partie I : (10 points)

Exercice 1 : (2,5 points)

Soient les déclarations suivantes :

Type
TEmploye = Enregistrement Mat : Chaîne[30] Age : Octet Genre : caractère
Fin TEmploye
FEmp = fichier de TEmploye
FEnt = fichier d'entiers

Objet	Type / Nature	Rôle
F	FEnt	Fichier d'entiers
F1	FEmp	Fichiers des employés
F2	Texte	Fichier texte
E	TEmploye	Enregistrement d'un employé
Ch	Chaîne	Chaîne de caractères

En se basant sur les déclarations précédentes, dire pour chacune des instructions suivantes si elle est correcte ou non. Dans le cas où elle ne l'est pas, justifier votre réponse.

- a) Lire (F1,E)
- b) Ecrire(F, Ch)
- c) Pointer (F2,4)
- d) Ecrire(F1,E.Age)

Exercice 2 : (4 points)

On considère une suite U définie par :

$$\begin{cases} U_1 = 1 \\ U_2 = 2 \\ \forall n \geq 3, U_n = U_{n-1} + k * U_{n-2} \text{ avec } k \text{ un entier donné } (k > 0) \end{cases}$$

- a) Quel est l'ordre de récurrence de cette suite ? Justifiez votre réponse.
- b) Ecrire une analyse d'une fonction qui vérifie si un entier p donné est un terme de la suite U ou non. Si p est un terme de la suite U , la fonction retourne le rang r de p , sinon, elle retourne -1.
- c) En déduire l'algorithme correspondant de cette fonction.

Exercice 3 : (3,5 points)

Ecrire une analyse d'une procédure intitulée **Convert_Bin_Hex** permettant de lire un nombre binaire puis de le convertir en hexadécimal, selon le principe suivant :

- Regrouper les chiffres binaires par bloc de 4, en partant de la droite.
- Si nécessaire, compléter le dernier bloc obtenu, par des 0 à gauche, pour former un bloc de 4 chiffres binaires.
- Convertir chaque bloc en hexadécimal.

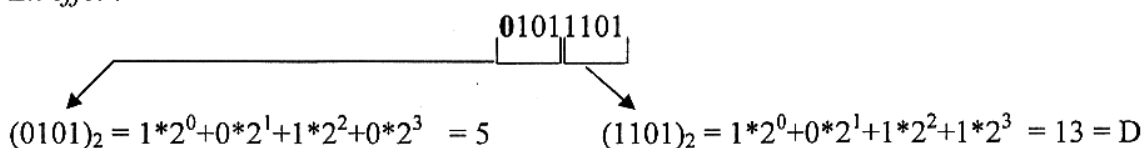
NB : la conversion d'un bloc de 4 chiffres binaires en décimal est supposée faite par une fonction **conv_bloc**, que le candidat n'est pas appelé à écrire et dont l'entête est la suivante :

DEF FN conv_bloc (bloc : entier) : chaîne

Exemple:

La conversion du nombre binaire 1011101 en hexadécimal est : $(5D)_{16}$

En effet :



Pour cet exemple : **conv_bloc (1101)** renvoie la valeur "D" et **conv_bloc (0101)** renvoie la valeur "5"

Partie II : (10 points)

Pour augmenter l'audience de ses émissions, une chaîne de télévision a organisé un jeu qui consiste à demander à **n** téléspectateurs de rappeler le classement des quatre pays demi-finalistes de la dernière coupe du monde de football 2006.

Les quatre pays demi-finalistes sont :

- L'Allemagne, représentée par la lettre "A"
- L'Italie, représentée par la lettre "I"
- Le Portugal, représenté par la lettre "P"
- La France, représentée par la lettre "F"

Chaque téléspectateur est appelé à envoyer par SMS, sa proposition de classement en mettant les lettres représentant chacun des pays demi-finalistes selon leur classement à la fin de la coupe du monde 2006. Ainsi, le texte d'un SMS n'est considéré valide que s'il contient une chaîne formée exactement des quatre lettres "A", "I", "P" et "F" et que chacune d'elles n'apparaît qu'une seule fois dans cette chaîne.

N.B : le programme doit accepter les lettres majuscules et minuscules.

Exemples :

- Si le téléspectateur envoie la chaîne "PPIF", son SMS ne sera pas valide car il ne contient pas la lettre "A".
- Si le téléspectateur envoie la chaîne "APAIF", son SMS ne sera pas valide car il contient plus de quatre lettres.
- Si le téléspectateur envoie la chaîne "iPaF", son SMS sera valide et sa proposition de classement est :
 1. Italie (I)
 2. Portugal (P)
 3. Allemagne (A)
 4. France (F)

On se propose d'écrire un programme qui permet de gérer ce jeu télévisé.

Pour cela ce programme devra permettre :

- 1) de saisir les textes des SMS des n téléspectateurs ($n \leq 5000$) ainsi que leurs numéros de téléphone. Les textes des SMS valides ainsi que les numéros des téléphones correspondants seront sauvegardés dans un fichier "**C:\SMS.dat**".

N.B : Le candidat n'est pas appelé à contrôler la saisie des numéros de téléphone.

- 2) de former à partir du fichier "**SMS.dat**", un tableau d'enregistrements intitulé **Foot**. Chaque enregistrement de ce tableau doit contenir les informations suivantes :
 - Le **numéro** de téléphone d'un participant
 - Le **classement** proposé par ce participant, c'est-à-dire les 4 lettres qui constituent son SMS
 - Le **score** de ce participant, qui est déterminé en fonction de la proposition du téléspectateur comparée à la chaîne gagnante. La chaîne gagnante étant "**IFAP**" puisque le classement à la fin de cette coupe du monde est le suivant :
 1. Italie (**I**)
 2. France (**F**)
 3. Allemagne (**A**)
 4. Portugal (**P**)

Le score est calculé de la façon suivante :

- **100** points si les quatre lettres sont à la bonne position
 - **50** points si seulement deux lettres sont à la bonne position
 - **25** points si seulement une seule lettre est à la bonne position
 - **0** point si aucune lettre n'est à la bonne position.
- 3) de créer un tableau **NT** contenant les numéros de téléphone (sans redondance) des participants ayant le score le plus élevé.

N.B : *Un téléspectateur peut envoyer plus d'un SMS pour augmenter ses chances de gagner.*

- 4) d'afficher le nom et le prénom du (ou des) gagnant(s), sachant que le nom et le prénom d'un téléspectateur, correspondant à un numéro de téléphone donné, sont supposés obtenus à partir d'un fichier nommé "**abonnes.dat**" fourni par une société de télécommunication, et dont la structure est la suivante :
 - numéro de téléphone,
 - nom
 - prénom

N. B : *Le candidat n'est pas appelé à saisir le contenu du fichier "abonnes.dat".*

Travail demandé

1. Analyser le problème en le décomposant en modules et déduire l'algorithme du programme principal qui permet de réaliser le traitement décrit précédemment.
2. Analyser chacun des modules envisagés précédemment et en déduire les algorithmes correspondants.