



مكتب التكوين المهني وإنعاش الشغل

Office de la Formation Professionnelle  
et de la Promotion du Travail

Direction Recherche et Ingénierie de Formation

Examen de passage

Session Juin 2010

Filière : TDI

Epreuve : Pratique

Niveau : Technicien Spécialisé

Durée : 4 h 30

Barème : 20 Pts

**Important : assurez-vous que tous les éléments de vos projets sont sauvegardés dans un dossier portant votre nom et prénom et le numéro de la variante de l'examen comme suit "NOM PRENOM Variante".**

### Variante n° 2

#### **PARTIE I : PROGRAMMATION STRUCTUREE EN C OU C++ (4 Pts)**

1. La Direction d'une entreprise désire automatiser le calcul de l'indemnité à verser aux cadres en cas de licenciement.

Après un an d'ancienneté dans l'entreprise, il sera alloué aux cadres licenciés une indemnité tenant compte de leur ancienneté et s'établissant comme suit :

- la moitié du salaire d'un mois par année d'ancienneté : pour la tranche d'ancienneté entre 1 et 10 ans
- au delà de 10 ans un mois de salaire par année d'ancienneté
- une indemnité supplémentaire serait allouée aux cadres âgés de plus de 45 ans de :
  - 2 mois si le cadre est âgé de 46 à 49 ans
  - 5 mois si le cadre est âgé de plus de 50 ans.

Ecrire un programme qui permet de saisir l'âge, l'ancienneté et le dernier salaire et d'afficher l'indemnité du cadre. **(1.5 pts)**

2. Ecrire un programme permettant de trier un tableau de n réels dans un ordre croissant en utilisant la méthode de tri bulle **(1 pt)**

3. Ecrire un algorithme qui effectue la lecture d'une matrice carrée A ainsi que sa taille n et affiche la trace de A. **(1 pt)**

(Pour une matrice  $A(a_{i,j})$ ,  $\text{Trace}(A) = \sum a_{i,i}$  la somme des éléments sur la diagonale).

4. Ecrire un programme qui demande à l'utilisateur de saisir un entier N et qui affiche la figure suivante sous forme des étoiles. **(0.5 pt)**

**N=3**  
**\*\*\***

\*\*  
\*

## **PARTIE II : PROGRAMMATION ORIENTEE OBJET EN VB.NET, C# OU JAVA (10 Pts)**

On souhaite développer une application pour la gestion des modules enseignés par des professeurs dans une Ecole privée.

Soit la classe **module** qui modélise un module et qui comportera les attributs suivants :

reference  
intitule  
professeur (matricule du Professeur qui enseigne ce module)

Soit la classe **Professeur** qui modélise un Professeur et qui comportera les attributs suivants :

matricule  
nom  
prenom  
echelle  
listeModule

### **Travail à faire :**

- 1- Codage de la classe Module
  - a. Ecrire la classe Module avec un compteur des objets créés. **(1pt)**
- 2- Codage de la classe Professeur
  - a. Ecrire la classe Professeur avec un compteur des objets créés. **(1pt)**
- 3- Ajouter la méthode compare qui teste l'égalité de 2 objets Professeur (2 objets Professeur sont égaux s'ils ont le même matricule). **(1pt)**
- 4- Créer une exception nommée erreurEchelle qui se déclenche lors de la saisie d'une échelle de professeur qui n'est pas comprise entre 8 et 11. **(1pt)**
- 5- Ajouter une méthode choixProfesseur qui désigne l'enseignant d'un module. **(1pt)**
- 6- Ajouter une méthode supprimerProfesseur qui annule le choix de l'enseignant d'un module. **(1pt)**
- 7- Ajouter une méthode ajouterModule qui ajoute un module à la liste des modules pris en charge par un professeur. **(1pt)**
- 8- Ajouter une méthode supprimerModule qui supprime un module de la liste des modules d'un professeur. **(1pt)**
- 9- Ecrire un programme de test qui crée 4 professeurs et 5 modules : **(1pt)**
- 10- Ecrire un programme permettant d'enregistrer les objets créés dans deux fichiers texte : Professeurs.txt et Modules.txt **(1pt)**

**PARTIE III : PROGRAMMATION EVENEMENTIELLE EN VB.NET, C# OU JAVA (6 PTS)**

Pour plus de convivialité, on souhaite développer quelques fonctionnalités de l'application de gestion de la répartition des modules en mode graphique.

- 1- Créer une interface qui permet la gestion des modules.
  - a. Ajouter un module **(0.5pt)**
  - b. Supprimer un module selon sa référence. **(0.5pt)**
  - c. Modifier un module selon sa référence. **(0.5pt)**
  - d. Rechercher un module selon sa référence ou son intitulé. **(1pt)**
  - e. Afficher dans un label le nombre des modules créés **(0.5pt)**
  
- 2- Créer une interface qui permet la gestion des professeurs.
  - a. Ajouter un professeur. **(0.5pt)**
  - b. Supprimer un professeur selon son matricule. **(0.5pt)**
  - c. Modifier les informations d'identité d'un professeur. **(0.5pt)**
  - d. Rechercher un professeur selon son matricule. **(0.5pt)**
  - e. Afficher l'ensemble des professeurs dans une grille **(0.5pt)**
  
- 3- Créer un menu permettant d'accéder à ces différentes fonctionnalités. **(0.5pt)**