



<http://www.ista-ntic.net/>

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 Juillet 2012

Filière : Techniques de Développement Multimédia

Epreuve : Pratique – V3-2

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° 8

DOSSIER 1 (7,5 Pts)

N.B. - Veillez à sauvegarder chacune des questions de ce dossier dans un fichier à part.

- Les fichiers que vous manipulerez dans ce dossier se trouvent dans le répertoire V32.

Exercice 1 (4 pts)

On considère les images « imageV32_a.jpg », « imageV32_b.jpg », « imageV32_c.jpg » et « imageV32_d.jpg » qui représentent respectivement les photos de : une assiette de fruits vide, une orange, une pomme et une cerise.

1- Remplir l'assiette « imageV32_a.jpg » par les fruits suivants, en veillant à sélectionner, redimensionner et faire les rotations nécessaires des images : (3 pts)

- 3 oranges.
- 2 pommes.
- 4 cerises.

2- Modifier la couleur, en vert, des 2 pommes. (1 pt)

Exercice 2 (3,5 pts)

On considère les fichiers son « sonV32_a.wma » et « sonV32_b.wav » qui vous sont fournis en fichiers joints.

1- Le fichier « sonV32_a.wma » représente l'enregistrement d'un petit discours. Convertir ce fichier en .wav, et échantillonner les 2 fichiers « sonV32_a.wav » « sonV32_b.wav » à la fréquence maximale. (1 pt)

2- Limiter les durées des 2 fichiers son à 40 secondes, et créer 4 régions (reg1, reg2, reg3 et reg4), d'une durée de 10 secondes chacune, au niveau du fichier « sonV32_b.wav », et enfin créer une playliste associée à ce fichier qui ne comporte que les régions 2 et 3. (1,5 pt)

3- Normaliser et étirer les durées des 2 fichiers son, à 50 secondes. Le fichier « sonV32_b.wav » représente une musique douce. Appliquez-la comme musique de fond au fichier « sonV32_a.wav ». (1 pt)

DOSSIER 2 (4,5 Pts)

Exercice 1 (1,5 pts)

<http://www.ista-ntic.net/>

Une matrice M carrée (nombre de ligne égale le nombre de colonnes) est dite **diagonale supérieure** si les valeurs se situant **au dessous** de la diagonale principale sont nulles (voir exemple ci-dessous).

Ecrire un programme, dans un langage de programmation structurée, permettant de construire une matrice diagonale supérieure de degré n (n lignes, n colonnes), dont les valeurs **au dessus** de la diagonale sont constituées de la manière suivante (les valeurs en dessous de la diagonale sont évidemment nulles) :

$M[i,j] = 2*i + j$ avec i le numéro de la ligne et j le numéro de la colonne.

Exemple : Pour $n = 3$, on obtient :

```
M = 3 4 5
    0 6 7
    0 0 9
```

Exercice 2 (3 pts)

On considère une phrase comme étant une liste de mots (chaînes de caractères) séparés par un ou plusieurs espaces.

On appellera cette phrase « phrase dernière » si chaque mot de cette phrase se termine par la même lettre, et dans le cas contraire, on l'appellera « phrase non dernière ».

Ecrire un programme, dans un langage de programmation structurée, qui contient une fonction permettant de lire une phrase saisie par l'utilisateur, et une autre fonction permettant de vérifier si cette phrase est dernière ou non et d'afficher dans chaque cas le message qu'il faut.

Exemples :

- La phrase : « une chaîne constituée de ce caractère » est dernière.
- La phrase : « Cette phrase est non dernière » est non dernière.

DOSSIER 3 (8 Pts)

N.B. Les programmes seront écrits dans un langage de programmation orientée objet de votre choix.

Un gérant d'un complexe résidentiel souhaite gérer les différentes maisons de son bien ainsi que le loyer reçu chaque fin de mois de la part des locataires.

Chaque maison est caractérisée par :

- Un code de type entier,
- Une superficie en m^2 de type entier,
- Un style de maison (aire+1, aire+2 ou aire+3),
- Un nom du locataire de type chaîne de caractères,
- Un montant de loyer de type réel.

1. Créer la classe « maison ». (1 pt)
2. Ajouter à cette classe les sélecteurs et les modificateurs ainsi qu'un constructeur avec paramètres. (0,5 pt)
3. Ajouter la méthode toString() pour afficher les informations d'une maison. (0,5 pt)

<http://www.ista-ntic.net/>

Le complexe résidentiel est caractérisé par :

- Un nom.
 - Un tableau de maisons.
 - Une adresse.
4. Créer la classe « complexe » et un constructeur avec paramètres. (1 pt)
 5. Créer une méthode permettant d'ajouter une maison au complexe. (0,25 pt)
 6. Créer une méthode permettant de supprimer une maison du complexe. (0,25 pt)
 7. Ecrire une méthode renvoyant le code de la maison ayant le loyer le plus bas. (0,75 pt)
 8. Ecrire une méthode permettant de trier les maisons selon la superficie dans un ordre croissant. (1 pt)
 9. Ecrire une classe test afin de tester le code déjà écrit. (0,5 pt)
 10. Créer une interface permettant d'ajouter une maison au complexe. Effectuer les contrôles nécessaires (Types de champs, 3 aires seulement...). (1,5 pt)
 11. Créer une interface permettant d'afficher les maisons d'un même style (aire+...). (0,75 pt)

<http://www.ista-ntic.net/>