

**EXAMEN FINAL – 2<sup>e</sup> SEMESTRE**

**Niveau : LICENCE**

**Filières : Tronc commun**

**EPREUVE : CSC 242 - Algorithmes & Programmation Structurée avec Python**

DUREE : 3 HEURES

Chargé du Cours : M. SHABAN

**Exercice 1 (6 pts)**

L'entreprise PressingLux offre des services d'entretiens auxquels s'abonnent les clients. Voyant sa clientèle grandissante, elle désire numériser son annuaire. L'annuaire est un répertoire mis à jour chaque année. Il rassemble les informations suivantes sur chaque client : nom, adresse ou domicile, coordonnées téléphoniques, date du premier abonnement. Avec le répertoire numérique, les opérations comme : consulter les informations d'un client, rechercher un client à partir de son nom, mettre à jour les coordonnées téléphoniques d'un client seront moins pénibles qu'avec le répertoire physique.

Faites donc la déclaration en Python de la classe enveloppe d'un dictionnaire nommée « **Annuaire** » qui permettrait de numériser le répertoire de PressingLux. Dans la déclaration, ajoutez les méthodes nécessaires pour effectuer spécifiquement les opérations précédemment mentionnées.

**Indications :**

Pour déclarer une classe enveloppe d'un dictionnaire, il faut avoir dans la classe qu'un attribut de type dictionnaire. Vous n'avez pas besoin de faire de l'encapsulation lors de l'implémentation de cette classe. Pensez à rédiger un programme simple pour tester la classe « Annuaire ».

**Exercice 2 (11 pts)**

Une université veut améliorer la gestion de sa médiathèque. Après analyse des tâches à mener dans le projet, l'on vous demande de réaliser un système de gestion

des emprunts. Les interviews avec le gérant de la médiathèque ont permis de résumer les différents besoins sous forme de phrase suivante :

- La médiathèque enregistre chaque abonné en gardant les informations comme : matricule de la carte d'abonné, nom, prénoms, sexe, date d'abonnement (premier abonnement à la médiathèque).
- Les ressources empruntées à la médiathèque sont de toutes sortes : livres (roman, bande dessinée, articles...), films, cassettes audio ou cassettes vidéo. Chaque ressource est enregistrée en plusieurs exemplaires dans la médiathèque. Ainsi pour enregistrer chaque exemplaire, le gérant enregistre la référence de la ressource, le numéro de l'exemplaire, le titre de la ressource, la description de la ressource, le type de la ressource (livre / film / audio).
- Le gérant enregistre pour chaque emprunt d'une ressource les informations suivantes : la référence (numéro d'identification) de la ressource, le numéro d'exemplaire, la date d'emprunt, le matricule de la carte de l'abonné et la date de retour de la ressource empruntée.
- Dans la médiathèque, quand un abonné ne trouve pas d'exemplaire d'une ressource, il peut faire une réservation. Ainsi dès qu'un exemplaire de la ressource est à nouveau disponible, le gérant remet immédiatement cet exemplaire à l'auteur de la première réservation effectuée sur cette ressource. Une réservation est semblable à un emprunt.
- Pour une réservation, le gestionnaire enregistre la référence de la ressource, la date de la réservation, le matricule de la carte d'abonné et l'état de la réservation qui par défaut prend la valeur « Non servi ».

Une fois qu'un abonné aura effectué un retour d'un exemplaire de la ressource réservée, le gestionnaire se charge de remettre l'exemplaire retourné à l'abonné dont la date de réservation est la plus ancienne. L'état de la réservation de cet abonné passera donc dans l'état « Servi ».

L'état de la réservation permet donc au gérant de savoir si oui ou non l'abonné ayant fait la réservation a déjà été servi.

Travail à faire :

1. En analysant ce résumé provenant des interviews avec le gérant de la médiathèque, identifier et lister toutes les classes (avec leurs attributs) nécessaires à la réalisation du système de gestion des emprunts.

2. En utilisant les principes de la programmation orienté objet, faites les déclarations de ces classes en langage algorithmique et en langage Python.  
Dans vos déclarations, utilisez le mode de protection adéquat sur les informations : matricule de la carte d'abonné, référence de la ressource, état de la réservation. Ces informations ne doivent en aucun cas être « public ».
3. Dans un programme principal (en Python ou en Algorithmique) bien commenté, implémentez le scénario suivant :
  - a. Deux nouveaux étudiants sont inscrits à la médiathèque : Paul TOTO et Pierre TOTO
  - b. Paul emprunte à la médiathèque un exemplaire du livre « Les algorithmes avec Python » (les retours des ressources se font après deux semaines depuis le jour de l'emprunt).
  - c. Pierre veut emprunter le film « Opération Fortune : Ruse de Guerre », mais plus aucun exemplaire n'est disponible. Il fait donc une réservation. Pour ce film, l'on compte déjà deux réservations faites avant celle de Pierre. La première faite par « Elodie » et la seconde faite par « Gaston ».
  - d. Enfin, un retour d'un exemplaire du film « Opération Fortune : Ruse de Guerre » est effectuée deux jours après la demande de Pierre. Montrez dans votre programme qu'au moins une réservation sera satisfaite !

#### Indications :

- Utilisez le principe de l'héritage et du polymorphisme dans vos déclarations pour améliorer la performance du futur système
- En appliquant le principe de l'encapsulation, tous les attributs seront publics sauf les informations suivantes : matricule de la carte d'abonné, référence de la ressource, état de la réservation.
- Les dates peuvent être déclarées comme : chaîne de caractère ou Date
- Lors de l'implémentation du scénario, vous pouvez définir des valeurs arbitraires seulement pour les informations manquantes. (Exemple : Les références des ressources n'ont pas été définies, vous pourrez donc leur attribuer des valeurs de votre choix)
- La lecture de vos programmes et algorithmes doit être facilitée par une bonne indentation et par une bonne utilisation des commentaires

### **Exercice 3 (3 pts)**

1. Énoncez et expliquez très brièvement les principes de la programmation orienté objet.
2. Vous avez une discussion avec un programmeur qui ne connaît que la programmation procédurale. Donnez-lui un avantage de la programmation orienté objet et illustrez cet avantage par un exemple.