



Site : ☐ Luminy ☐ St-Charles ☒ St-Jérôme ☐ Cht-Gombert ☐ Aix-Montperrin ☐ Aubagne-SATIS

Sujet de : ☐ 1^{er} semestre ☒ 2^{ème} semestre ☐ Session 2 Durée de l'épreuve : 2h

Examen de : L1 Nom du diplôme : Licence MPC

Code du module : SMP2U23J Libellé du module : Programmation

Calculatrices autorisées : NON Documents autorisés : NON

1 Attribut/Héritage/Agrégation et composition

Pour chacune des phrases suivantes, écrivez un code simple différent des exemples vus en cours et qui illustre au mieux cette phrase (votre exemple doit correspondre à une situation concrète de la vie réelle : par exemple une classe `Polygone` plutôt qu'une classe `A` n'ayant aucun sens) :

1. « Un attribut permet à deux objets de la même classe d'être différents »
2. « L'héritage permet à deux classes d'avoir des similitudes, mais aussi des spécificités »
3. « L'agrégation et la composition se distinguent sur la possibilité d'un objet à être associé une seule fois ou plusieurs fois à un autre objet »

2 Conversion de devise

On considère que 1 €(euro) vaut \$1.13 (dollar) et que \$1 (dollar) vaut £0.75 (livre sterling). On souhaite concevoir une classe `Conversion` dont le constructeur prend 3 paramètres :

- une chaîne de caractères correspondant à la devise source,
- une chaîne de caractères correspondant à la devise cible,
- un réel représentant le taux de conversion : si r est le taux, une unité de la devise initiale correspond à r unités de la devise cible.

Cette classe possède

- une méthode de conversion de la devise source vers la devise cible,
- une méthode renvoyant le convertisseur inverse (permettant de convertir depuis la devise cible vers la devise source),

Question 2.1. Donnez un diagramme UML de cette classe.

Question 2.2. Codez cette classe.

Question 2.3. Codez des tests pour cette classe.

Question 2.4. Donnez le code et les tests d'une fonction qui construit et renvoie un convertisseur d'une devise d_1 vers une devise d_3 à partir d'un convertisseur d_1 vers d_2 et d'un convertisseur d_2 vers d_3 .

Question 2.5. On souhaite remplacer la fonction précédente afin de pouvoir utiliser le code suivant :

```
conv_euro_vers_dollar = Convertisseur("euro", "dollar", 1.13)
conv_dollar_vers_livre_sterling = Convertisseur("dollar", "livres sterling", 0.75)
conv_euro_vers_livre_sterling = conv_euro_vers_dollar + conv_dollar_vers_livre_sterling
```

Écrivez le code complémentaire de la classe et les nouveaux tests.

3 « Objectifier »

Le code suivant a été conçu pour gérer le catalogue bibliothèque

```
def decrire_livre(livre):
    titre, prenom_auteur, nom_auteur, annee = livre
    return f'"{titre}" de {prenom_auteur} {nom_auteur} ({annee})'

def decrire_bibliotheque(bibliotheque):
    s = f"Bibliothèque avec {len(bibliotheque)} livre(s)\n"
    for livre in bibliotheque:
        s += f" - {decrire_livre(livre)}\n"
    return s

livre1 = ("Intro à l'algo", "André", "Cursif", 1990)
livre2 = ("La POO en Python", "Paul I.", "Morf", 2010)
livre3 = ("Les tests pour les nuls", "Deb", "Eugue", 2015)
ma_bibliotheque = [livre1, livre2, livre3]
print(decrire_bibliotheque(ma_bibliotheque))
```

Question 3.1. Écrire une fonction `rechercher(titre, bibliotheque)` qui prend en arguments une chaîne de caractères `titre` et une bibliothèque comme dans l'exemple ci-dessus et renvoie l'indice du livre dont le titre est `titre` dans la bibliothèque s'il existe, et `None` sinon.

Question 3.2. Écrire une fonction `trier(bibliotheque)` qui prend en argument une bibliothèque comme dans l'exemple ci-dessus et en renvoie une version dans laquelle les livres ont été triés par ordre alphabétique de nom d'auteur. On pourra supposer qu'on dispose d'une fonction `tri_chaines(liste_de_str)` qui trie une liste de chaînes de caractères par ordre alphabétique et renvoie le résultat.

Question 3.3. On souhaite remplacer le code précédent par une version orientée objet. Réécrivez l'ensemble du code (y compris les fonctions `rechercher` et `trier`) en suivant ce style de programmation du mieux possible.

Question 3.4. Écrivez des tests pour le nouveau code orienté objet.

Question 3.5. On souhaite maintenant gérer le catalogue d'une médiathèque proposant des livres, des CD et des DVD. Proposez une modélisation UML dans la continuité de votre code objet précédent. N'hésitez pas à faire preuve de bon sens et d'imagination pour enrichir votre réponse afin qu'elle soit assez réaliste. Il n'est pas demandé de coder la solution.