

Programmation orientée objet Java

Examen

Durée : 1h30

Exercice 1 [5 pts]

1. Quels sont les avantages d'utilisation des exceptions en Java ?
2. C'est quoi une classe générique ? Donner un exemple.
3. Expliquer le mécanisme d'insertion dans une collection `HashSet` et donner sa complexité en temps d'exécution.
4. Citer une différence entre les classes `Iterator` et `ListIterator`.
5. Donner la complexité de recherche d'un élément dans un `TreeSet` de taille n . Justifier votre réponse.

Exercice 2 [5 pts]

On souhaite créer un itérateur sur les tableaux.

```
public class TabIter<E> implements Iterator<E>
{
    private E[] tableau;
    ...
}
```

On rappelle qu'un itérateur fournit des méthodes pour parcourir des éléments :

- `boolean hasNext()` pour savoir s'il reste des éléments à parcourir,
- `E next()` pour obtenir le prochain élément à parcourir.

Implémenter ces méthodes dans `TabIter`. On utilisera un champ `index` qui stockera la position courante dans le tableau. Le constructeur prend deux paramètres : le tableau et l'index. On prendra comme convention que dès qu'un pointeur `null` est rencontré, il ne reste plus d'éléments à parcourir. La taille d'un tableau `t` est `t.length`. Tester le bon fonctionnement de la classe `TabIter` en utilisant des tableaux de type `Integer` et `String`.

Exercice 3 [10 pts]

Un utilisateur est connecté sur un site de commerce électronique. L'utilisateur a un identifiant, un nom, un login et un mot de passe. Nous souhaitons gérer ses achats en ligne. Chaque produit a un identifiant, un prix unitaire hors taxe (`prixHT`) et la quantité choisie par l'utilisateur. Les produits à acheter sont ajoutés au panier de l'utilisateur. La classe `Produit` dispose d'une méthode abstraite `prix()`. Le prix est calculé en fonction de la catégorie du produit :

- Produits de catégorie A : application d'une taxe de 16% au prix total
- Produits de catégorie B : application d'une taxe de 20% au prix total

Dans un panier (classe `Panier`) les produits choisis par l'utilisateur sont représentés sous forme d'un `TreeSet`.

1. Proposer un diagramme de classe.
2. Écrire les classes Java en ajoutant les constructeurs et les autres méthodes jugées nécessaires. Les getters et setters sont demandés uniquement pour la classe `Panier`. Le constructeur de la classe `Utilisateur` lève une exception `IncorrectPass` si le mot de passe contient moins de 5 caractères.
3. Ajouter à la classe `Panier` une méthode `totalPaiement()` qui renvoie le montant total.
4. Écrire une classe `Test` avec une méthode `main()` qui teste la création d'un panier et l'affichage du total à payer. L'exception `IncorrectPass` est gérée dans `main()`.