

Devoir maison POO

Gestion de pharmacie

Une pharmacie souhaite traiter les informations concernant son stock de médicaments par ordinateur.

On vous propose de représenter les informations concernant le **Stock** de médicaments sous forme d'un **tableau d'index** et d'un **ensemble de liste linéaire chaînée**.

Le tableau d'index contient 26 éléments (au nombre des lettres de l'alphabet). Chaque élément du tableau contient deux informations:

- **car** : un caractère représentant la 1^{ère} lettre du nom du médicament
- **listeMed** : une référence vers une liste linéaire chaînée de médicaments ordonnés dont le nom commence par la lettre **car**

Le **Médicament** (un maillon) est décrit par un **nom** (attribut sur lequel la chaîne est ordonnée), la quantité disponible **nb** (nombre de boîtes dans le stock), le prix unitaire de l'achat **pua**, une marge bénéficiaire à la vente **marge** sous forme de pourcentage et un statut **remb** (booléen) indiquant si le médicament est remboursable ou pas.

La vente de médicaments à des clients s'effectue sur le comptoir de la pharmacie.

Le **Client** est décrit par son **nom** et **prénom**, sa date de naissance **dateN**, s'il est **assuré**, on lui rajoute le **NSS** (numéro de sécurité sociale).

Lors de la vente il faut prendre en considération les contraintes suivantes :

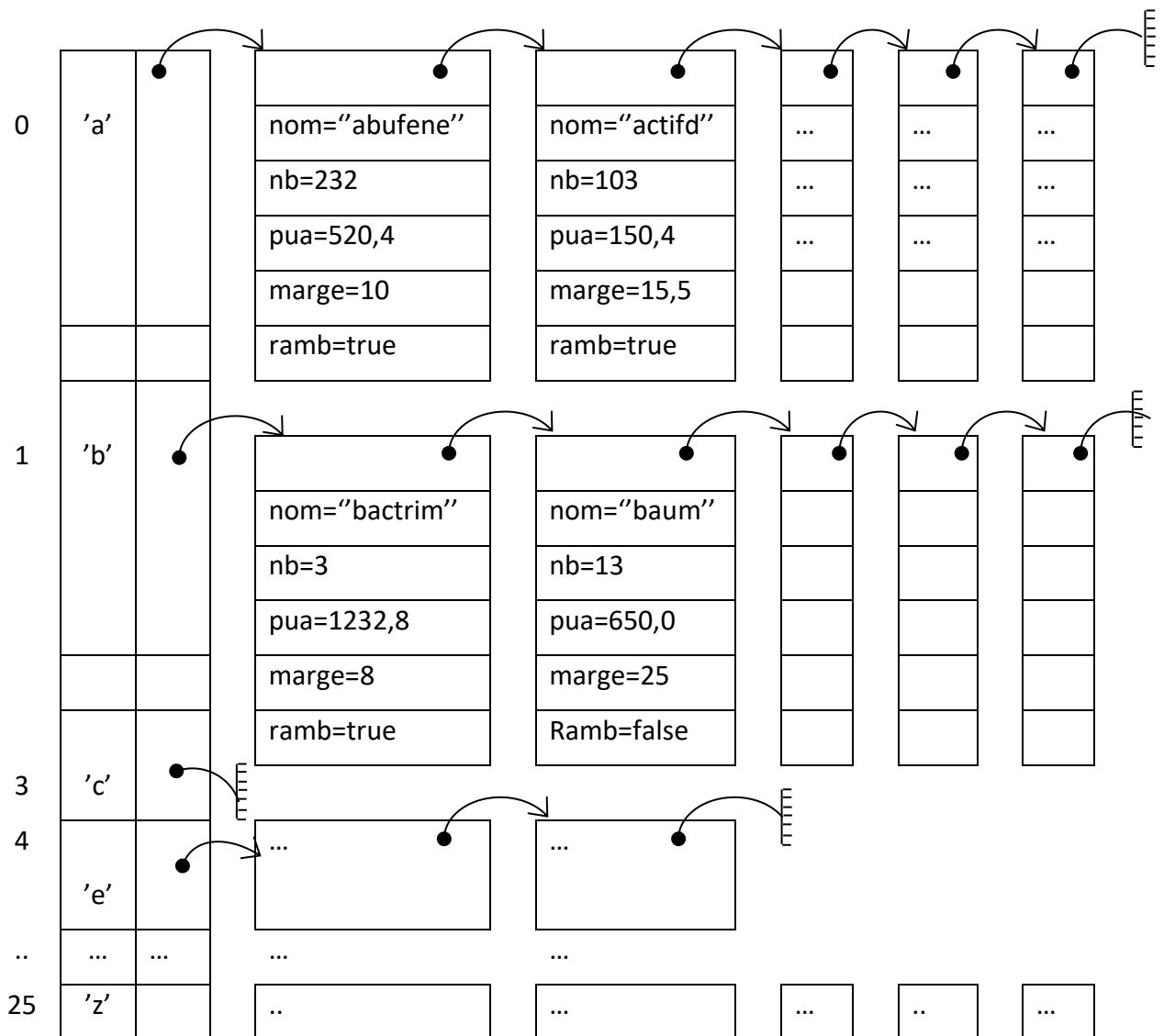
- le statut du médicament : remboursable ou pas (utilisé l'attribut **remb**)
- si le client n'est pas assuré, alors il paye la totalité du prix de vente du médicament,
- si le client est assuré, alors il paye 20% du prix de vente du médicament.

Travail à faire :

1. Définir les classes nécessaires à la représentation de ce stock (*voir le schéma ci-dessous*).
2. Écrire la méthode **rechercher(..)** permettant de rechercher un médicament par son nom dans le stock (accès direct dans l'index, puis séquentiel dans la liste)
3. Écrire la méthode **achat(..)** permettant au pharmacien d'alimenter son stock du médicament **med** (reconnu par son nom) ayant le prix d'achat **pa** par une certaine quantité **qte**. On considère qu'un médicament prend toujours le nouveau prix. Si le médicament n'existe pas, il faut l'insérer dans la bonne position.
4. Écrire la méthode **vente(..)** permettant au pharmacien de vendre un certain nombre **nb** de boîte d'un médicament **med** à un client **cl** et de retourner le montant à payer par le client. La vente n'est possible que si le nombre de boîte est disponible dans le stock. La vente impacte le stock du médicament en question. Si le nombre de boîtes du médicament vendu devient 0 alors le médicament doit être supprimé.
5. Écrire une méthode **valStock(..)** qui calcule et renvoie le montant total valorisant (au prix d'achat) les stock de médicament existant.

6. Créer une classe **Pharmacie** avec une méthode **main**, dans laquelle, on crée un nouveau stock, puis on l'alimente avec quinze (15) médicaments (des achats) et trois (03) clients. Simuler les cas suivants :

- Achat d'un médicament existant dans le stock.
- Vente d'un médicament remboursable à un client assuré
- Vente d'un médicament remboursable à un client non assuré
- Vente d'un médicament non remboursable à un client assuré
- Vente d'un médicament avec un stock limite (0 boîtes après la vente)



Remarque:

- Rajouter les constructeurs, les getters et les setters nécessaires.
- Les points entre les parenthèses des méthodes sont à remplacer par les paramètres formels
- L'utilisation de **LinkedList** et **ArrayList** de la bibliothèque standard java n'est autorisée
- Pour comparer deux chaînes de caractères, on peut utiliser la méthode **compareTo(..)** ou **compareToIgnoreCase(..)** de la classe **String**.