

Petit problème sur les listes simplement chaînées

Problème 0 : Gestion de hôtel du rail

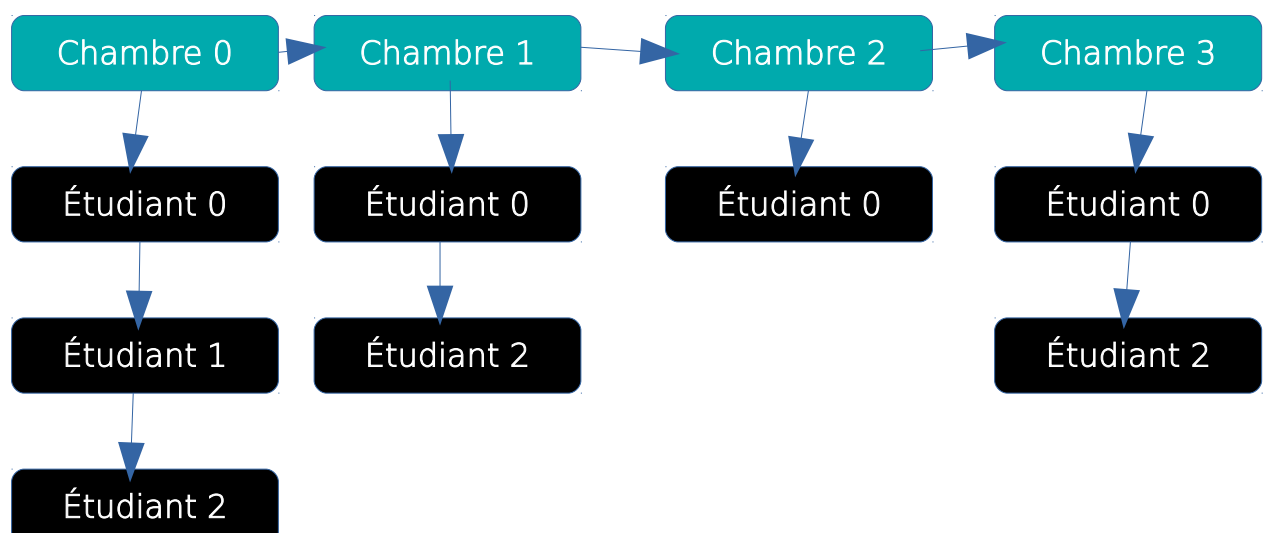
L'hôtel du rail veut un système d'information pour la gestion des chambres d'étudiants. Les chambres sont numérotées de **0 à n**. Chaque chambre peut avoir **quatre** étudiants au max.

- On veut pouvoir ajouter des étudiants dans une chambre donnée.
- Déplacer un étudiant d'une chambre à l'autre.
- Vérifier s'il y a de la place dans une chambre donnée.
- Vérifier s'il y a une place libre et dans quelle chambre.
- Compter le nombre d'étudiant dans l'hôtel.
- Compter le nombre d'étudiant dans une chambre donnée.
- Supprimer un étudiant s'il doit partir.
- Supprimer toute la base de donnée pour une nouvelle année universitaire.
- Effectuer un remplacement entre autre étudiant qui veut laisser ça place à un autre
- Dresser la liste des étudiants par chambre.
- Dresser la liste des étudiants d'une chambre donnée
- Supprimer une chambre et réaffecter les occupants
- Ajouter une chambre.

NB :

1. Créer une liste vide qui représentera l'hôtel
2. Ajouter quelque chambre vide par défaut
3. Le directeur de l'hôtel peut ajouter des chambres à tout moment
4. Une chambre peut être supprimer pour des réparations
5. Chaque chambre est une liste
6. Chaque étudiant est représenter par son **Nom, Prénom et Numéro étudiant**
7. Utiliser la modularité pour mieux organiser votre code
8. Proposer un **menu** pour faire tous ces opérations

Exemple de représentation hôtel à 4 chambres



Les structures pour vous aider

```
2  typedef struct Etudiant{
3      char *prenom;
4      char *nom;
5      int numeroEtudiant;
6      struct Etudiant *etudiantSuivant;
7  }Etudiant, *E;
8
9  typedef struct Chambre{
10     E premierEtudiant;
11     Etudiant *dernierEtudiant;
12     int numeroChambre;
13     struct Chambre *chambreSuivant;
14 }Chambre, *C;
15
16 typedef struct Hotel{
17     Chambre *premierChambre;
18     C dernierChambre;
19 }Hotel, *H;
```

Remarque :

```
Etudiant* == E
Chambre* == C
Hotel* == H
```

A vos claviers