

**Examen Administration des SGBD**

**M1 : IL-SII-IV**

**Enoncé :**

On s'intéresse à la gestion des services douaniers. Ci-après le schéma simplifié de la base de données :

**Brigade (Num\_Brig, Wilaya, NB\_Douanier, MatD\_Chef\*)**

**Douanier (MatD, NomD, PrenD, Date\_nais, AdrD, Ntel, Grade, Salaire, Fonction, Num\_Brlg\*)**

**Intervention (Num\_interv, Date\_interv, Lieu\_Inter, Num\_Brlg\*)**

**Participer (Num\_interv\*, MatD\*)**

**Produit (CodeP, NomP, Qte\_maxi\_authorized, Prix\_unitaire)**

**Saisie (CodeP\*, Num\_interv\*, Qte\_saisie)**

Sachant que :

- Les clés primaires sont soulignées et \* désigne une clé étrangère.
- Le garde d'un Douanier peut être : «AGENT», «COMMANDANT» ou «SERGENT».
- «MatD\_chef» et «MatD» sont du même domaine.
- Le salaire d'un douanier  $\geq 50\ 000$  DA.

**Exercice 1 : Fonctions générales de SGBD et Optimisation (11 points)**

1. Décrire l'architecture interne d'un SGBD en précisant :

- le niveau externe,
- le niveau conceptuel,
- le niveau interne.

Illustrer chaque niveau par un exemple tiré de la base des services douaniers.

2. Supposant que la table Douanier est créée sans contraintes d'intégrité :

- a. écrire les commandes SQL permettant d'ajouter toutes les contraintes nécessaires.
- b. Utiliser le dictionnaire de données pour afficher les contraintes créées.

3. Exprimer, à l'aide de contraintes, la règle suivante :

« Le salaire d'un douanier ne peut pas être supérieur au salaire du chef de sa brigade. »

4. On souhaite créer une vue donnant, pour chaque brigade, le nombre total d'interventions effectuées.

a. Donner la commande SQL de création de la vue.

b. Indiquer les répercussions de cette création sur le catalogue système.

c. On souhaite mettre en place un contrôle d'accès strict pour les chefs de brigade. Un chef de brigade doit pouvoir consulter uniquement le nombre total d'interventions par brigade, sans avoir accès aux détails des interventions ni à la table *Intervention*.

c.1 Créer un rôle CHEF\_BRIGADE.

c.2 Retirer tout droit d'accès direct à la table *Intervention* pour ce rôle.

c.3 Autoriser ce rôle à consulter uniquement le nombre total d'interventions par brigade.

5. On cherche les interventions où du tabac a été saisi avec une quantité supérieure à 3 tonnes.

a. Donner la requête SQL correspondante.

b. Construire l'arbre algébrique non optimisé associé.

c. Générer l'arbre algébrique optimisé en appliquant les règles de transformation.

d. Calculer le coût d'exécution correspondant au plan optimisé de cette requête.

### Statistique sur les données :

Card(Brigade)= 100 tuples	Nombre de bloc de Brigade : 10
Card(Douanier)=1000 tuples	Nombre de bloc de Douanier : 100
Card(Intervention)= 10000 tuples	Nombre de bloc de Intervention : 1000
Card(Participer)=50000 tuples	Nombre de bloc de Participer : 5000
Card(Produit)=100 tuples	Nombre de bloc de Produit : 10
Card(Saisie)= 1000 tuples	Nombre de bloc de Saisie : 10
Val(Produit, NomP)=100	
Max(Qte_saisie)=10	Min(Qte_saisie)=1
FS(A>val)= Max(A)-val/Max(A)-Min(A)	

### Exercice 2 : Gestions des accès concurrents (09 points)

1. Considérez l'ordonnancement O1 suivant :

R1(Z) R3(X) W3(Z) W4(X) R1(X) W2(X) W2(V) W5(V) W5(Y) R1(Y).

- Dites si O1 est sérialisable ou non, en construisant le graphe de précédence.
- Si O1 est sérialisable, décrivez tous les ordonnancements en série qui sont équivalents à O1. Sinon, déterminez quel est le nombre minimal de transactions qui doivent être supprimées de O1 pour que l'ordonnancement résultant soit sérialisable. Précisez ces transactions.

2. Considérez l'ordonnancement O2 suivant :

R4(Y) W1(Z) R2(Y) W3(X) W1(Y) R1(X) R3(Z) C3 W5(Z) C2 W5(Y) C5 W4(Z) C1 R4(X) C4

- Donnez le scenario d'exécution de cet ordonnancement dans le cas du verrouillage à deux phases. Construisez le graphe d'attente de cet ordonnancement. Existe-t-il un deadlock ? Si oui, proposez une solution à ce problème.
- Si la stratégie wait-die est utilisée pour gérer les demandes de verrouillage. Quelle est l'exécution finale obtenue ?
- Dans un SGBD M, chaque transaction Ti est associée à un identifiant id(Ti), qui est un nombre entier. Le mécanisme de contrôle de concurrence de M fonctionne comme suit : dans un ordonnancement O, une transaction Ti est autorisée à effectuer une écriture sur un élément X après une transaction Tj si, et seulement si, id(Ti)>id(Tj). Autrement dit, le mécanisme n'interdit pas une écriture Wi(X) après une écriture Wj(X) lorsque l'identifiant de Ti est strictement supérieur à celui de Tj.  
De la même manière, une transaction Ti est autorisée à effectuer une lecture sur un élément X après une écriture réalisée par une transaction Th si, et seulement si, id(Ti)>id(Th). Autrement dit, le mécanisme autorise une lecture Ri(X) après une écriture Wh(X) lorsque l'identifiant de Ti est strictement supérieur à celui de Th.
  - Prouvez ou réfutez que tout ordonnancement accepté par le mécanisme de contrôle de concurrence de M est sérialisable par graphe de précédence.

Bon courage