

**Examen National du Brevet de Technicien Supérieur**

Session de avril-mai 2019

**Spécialité :** Génie Logiciel

**Epreuve :** Epreuve Pratique de SI et POO

**Durée :** 4h 00

**PARTIE 1 : Cas Pratique**

**Section 1 : Etude de cas : Université 25 pts**

On veut décrire les séances de travaux dirigés (TD) des unités de valeurs (UV) dans un département d'Informatique de l'université de Dschang par la relation suivante:

FAC (N°TD, salle, horaire, N°enseignant, nom\_enseignant, prénom\_enseignant, N°UV, nomUV, N°étudiant, nom\_étudiant, prénom\_étudiant, adresse\_étud, date\_ins.)

On considère les dépendances fonctionnelles suivantes :

N°TD → salle, horaire

N°enseignant → nom\_enseignant, prénom\_enseignant

N°UV → nomUV

N°étudiant → nom\_étudiant, prénom\_étudiant, adresse\_étud

(N°UV, N°étudiant) → date\_ins.

L'enseignement, dans ce département est divisé en unités de valeurs, chacune étant identifiée par un numéro.

Un étudiant s'inscrit à une ou plusieurs UV (six au maximum), et pour chaque UV, à un groupe de TD (N°TD). Les inscriptions dans les différentes UV sont indépendantes les unes des autres. On mémorise la date d'inscription de chaque étudiant à chaque UV (date\_ins.).

Il y a une séance de TD par semaine pour chaque UV. Chaque TD a lieu dans une salle donnée et à un horaire donné. Les groupes de TD sont numérotés 1, 2, 3, ... pour chaque UV.

Un enseignant assure un ou plusieurs groupes de TD d'une ou plusieurs UV. Un groupe de TD d'une UV est assuré toute l'année par le même enseignant, plusieurs enseignants pouvant se partager les différents groupes de TD d'une même UV.

- Quelles redondances et anomalies de mise à jour sont impliquées par cette relation FAC? (4pts)
- Établir un graphe minimal des dépendances fonctionnelles de FAC. (4pts)
- Quelle est la clé de la relation FAC? (4pts)
- Quelle est la forme normale de la relation FAC? (Justifiez votre réponse) (4pts)
- Transformer votre graphe des dépendances fonctionnelles en MCD (4pts)
- Proposer une bonne décomposition de FAC et préciser la forme normale des nouvelles Relations (Modèle logique de données relationnel, MLDR). (5pts)

**Section 2 : Administration de BD avec MYSQL (35pts)**

L'objectif de cette section est d'implémenter le modèle logique obtenu dans la section 1.

**Ecrire en SQL les requêtes ci-dessous :**

- Créer la table « Etudiant » avec toutes les contraintes (N°etudiant est clé primaire). (7pts)
- Créer la table « UV » avec toutes les contraintes. (7pts)
- Créer la table « inscrit » avec toutes les contraintes (N°etudiant et N°UV sont clé primaire et sont clés étrangères). (7pts)
- On souhaite ajouter à la table « inscrit » précédemment créer l'attribut note, écrire une requête SQL qui apporte cette modification (7pts)

- e. L'étudiant de numéro « E0012 » a eu 12.5 à l'UV de numéro « BD045 » au lieu de 10.5, écrire en SQL la requête qui met à jour cette note. (7pts)

## **PARTIE 2 : POO 40pts**

### **Section 1 10pts**

- a. Transformez le MCD de la question (c.) en diagramme de classe. La classe « étudiant » possède les méthodes « AjoutBD() » et « ModifNote(Code\_uv, Note) ». AjoutBD() ajoute un étudiant dans la base de données et ModifNote(Code\_uv, Note) modifie la note de l'étudiant dans l'UV. (5pts)
- b. Utilisez un AGL de votre choix pour créer votre diagramme de classe. (5pts)

### **Section 2 Conception et implémentation des JHM 15pts**

- a. Créer la classe étudiant en Java (10 pts)
- b. Implémenter en Java le formulaire qui permet d'enregistrer un étudiant dans la base de données. (5pts)

### **Section 3 : Interfaçage avec la BD 15 pts**

- a. Ecrire en Java le code permettant de se connecter et ajouter des données du formulaire (de la question a) dans la BD (7pts)
- b. Développez le formulaire qui permet de modifier la note d'un étudiant, à une UV. (8pts)