

# Bases de données: Examen

Pierre Senellart ([pierre.senellart@ens.fr](mailto:pierre.senellart@ens.fr))

29 mai 2024

Les seuls documents autorisés sont 2 feuilles recto-verso (4 pages), avec le contenu de votre choix, lisible à l'œil nu. Cet examen dure deux heures et est constitué d'un unique problème de 11 questions, noté sur 20 points.

On souhaite modéliser le système d'informations des ressources humaines d'une entreprise. L'entreprise est organisée en services, nommés, chaque service ayant un responsable, qui est membre de ce service. Un service peut être décomposé en sous-services. Une personne donnée est nécessairement membre d'au moins un service, et peut avoir une fonction dans chacun des services auxquels elle appartient. Pour chaque personne, son nom, son numéro de téléphone éventuel, et son salaire sont également conservés. Toute personne a un responsable, qui n'est pas nécessairement le responsable de son service ; le chef d'entreprise est la seule personne dont le responsable est elle-même.

1. (2 points) Proposer un schéma entité–association pour ce système d'informations.
2. (2 points) En déduire un schéma relationnel et l'exprimer avec des ordres SQL `CREATE TABLE` ; y faire figurer les clefs primaires (`PRIMARY KEY`), clefs candidates (`UNIQUE`), clefs étrangères `REFERENCES`, contraintes `CHECK`, et indiquer quels attributs sont nécessairement `NOT NULL`.
3. (1 point) Quelles sont les contraintes logiques du système d'informations qui ne sont pas impliquées par les contraintes SQL ? Les exprimer sous forme de formule de la logique du premier ordre, si possible sous forme de TGD ou d'EGD.
4. (3 points) Écrire une requête calculant la liste des personnes dont le responsable n'est pas le responsable d'un service auquel elles appartiennent :
  - a) en algèbre relationnelle ;
  - b) en calcul relationnel ;
  - c) en SQL.
5. (2 points) Proposer trois plans d'exécution pour cette requête, et discuter du plan le plus efficace, en fonction des tailles des différentes relations.
6. (1 point) Donner une borne supérieure sur la complexité asymptotique, en les données, de l'évaluation de cette requête.
7. (2 points) Écrire en SQL une requête calculant le salaire moyen par service, pour les employés directement rattachés à ce service (pas à un sous-service).

8. (4 points) On souhaite vérifier qu'il n'y a pas de cycle dans la relation de responsabilité entre personnes, c'est-à-dire qu'il n'existe pas deux personnes  $x_0$  et  $x_n$  distinctes telles qu'il existe une suite de personnes  $x_1, \dots, x_{n-1}$  avec  $x_0$  responsable de  $x_1$ ,  $x_1$  responsable de  $x_2$ ,  $\dots$ ,  $x_{n-1}$  responsable de  $x_n$  et  $x_n$  responsable de  $x_0$ . Exprimer une requête booléenne testant la non-existence de tel cycle :
- a) en Datalog avec négation ;
  - b) en logique du premier ordre à point fixe (inflationniste ou non-inflationniste, au choix) ;
  - c) en logique du second ordre ;
  - d) en SQL.
9. (1 point) On souhaite utiliser la base de données ainsi construite pour concevoir un annuaire en ligne des employés, mais cet annuaire ne doit faire figurer ni le salaire des employés, ni leur responsable. Proposer une solution permettant de garantir que ces informations ne seront pas accessibles par le développeur de l'annuaire Web. Inclure les ordres SQL nécessaires à cette solution.
10. (1 point) On souhaite permettre aux utilisateurs de l'annuaire Web de mettre à jour leur numéro de téléphone. Discuter d'une stratégie possible pour ce faire, toujours en supposant que le développeur de l'annuaire Web ne puisse pas avoir accès aux informations de salaire et de responsable.
11. (1 point) Illustrer par un exemple concret un problème de sécurité potentiel provenant d'une mauvaise implémentation de l'annuaire Web, et discuter de comment empêcher une telle faille.