

## Tables

```
CREATE TABLE Ecole(  
  n_ecole INT,  
  nom VARCHAR(100),  
  adresse VARCHAR(100),  
  nomDirecteur VARCHAR(100),  
  
  PRIMARY KEY(n_ecole)  
)  
  
CREATE TABLE Enfant(  
  n_enfant INT,  
  nom VARCHAR(100),  
  prenom VARCHAR(100),  
  date_naissance DATETIME,  
  classe VARCHAR(100),  
  
  n_ecole INT,  
  
  PRIMARY KEY(n_enfant),  
  FOREIGN KEY(n_ecole) REFERENCES Ecole  
)  
  
CREATE TABLE Aliment(  
  n_aliment INT,  
  nom VARCHAR(100),  
  famille VARCHAR(100),  
  origine VARCHAR(100),  
  
  PRIMARY KEY(n_aliment)  
)  
  
CREATE TABLE Allergie(  
  n_enfant INT,  
  n_aliment INT,  
  symptomes VARCHAR(100),  
  traitement VARCHAR(100),  
  
  PRIMARY KEY(n_enfant, n_aliment),  
  FOREIGN KEY(n_enfant) REFERENCES Enfant,  
  FOREIGN KEY(n_aliment) REFERENCES Aliment,  
)
```

---

## Requêtes

```
!R1  
SELECT E.nom  
FROM Enfant E, Allergie A  
WHERE E.n_enfant = A.n_enfant  
      AND A.symptomes = 'eczema'  
      AND A.traitement = 'cortisone'  
      AND E.classe = 'CP' ;  
  
!R2  
SELECT E.nom  
FROM Enfant E, Allergie All, Aliment A  
WHERE All.n_aliment = A.n_aliment  
      AND E.n_enfant = All.n_enfant  
      AND A.nom = 'lait'
```

AND All.traitement = 'hospitalisation' ;

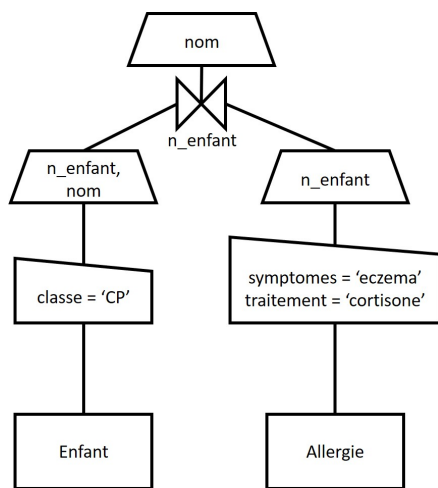
!R3

```
SELECT Ec.n_ecole
FROM Ecole Ec, Enfant E, Aliment
A, Allergie All
WHERE Ec.n_ecole = E.n_ecole
      AND E.n_enfant = All.n_enfant
      AND All.n_aliment = A.n_aliment
      AND A.nom = 'lait'
      AND E.classe = 'CP' ;
```

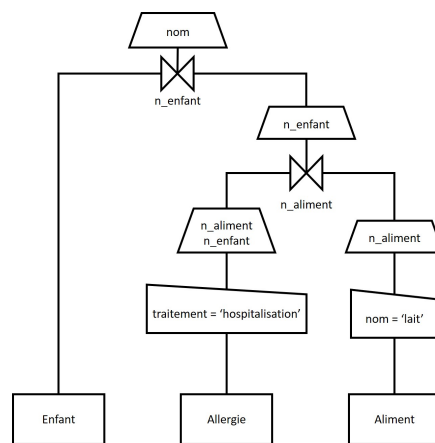
**Exercice 1 :** Pour chacune des 3 requêtes *R1*, *R2* et *R3*, répondez aux questions suivantes :

1. Après optimisation algébrique, donnez les arbres algébriques optimisés envisageables pour la requête en faisant apparaître les différents blocs d'opérations nécessaires (restriction/projection ou jointure/projection).

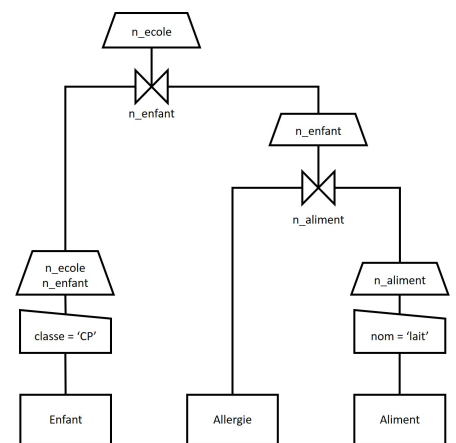
*R1 :*



*R2 :*

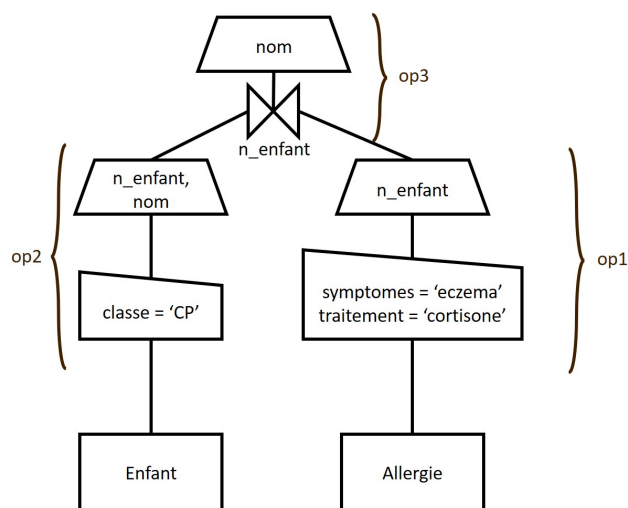


*R3 :*



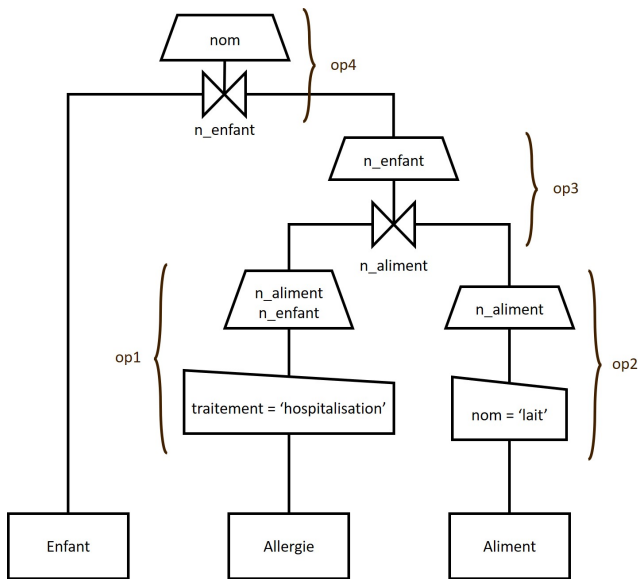
2. Décrivez en détail les différents plans d'exécution possibles pour la requête en supposant que le SGBD dispose de 2 algorithmes de restriction/projection : par balayage séquentiel ou par index et de 2 algorithmes de jointure/projection : par boucle imbriquée et par tri fusion.

*R1 :*



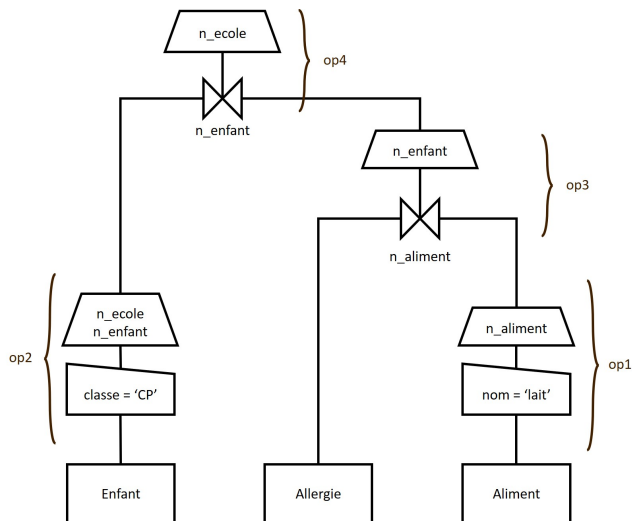
- *op1* : Restriction/projection -> RBS (Restriction par Balayage Séquentiel), pas d'index sur l'attribut classe.
- *op2* : Restriction/projection -> RBS
- *op3* : Joint/projection -> JBI (Joint par Balayage Index) et JTF (Jointure par Tri Fusion)

*R2 :*



- *op1* : Restriction/projection -> RBS (Restriction par Balayage Séquentiel), pas d'index sur l'attribut *classe*.
- *op2* : Restriction/projection -> RBS
- *op3* : Joint/projection -> JBI (Joint par Balayage Index) et JTF (Jointure par Tri Fusion)
- *op4* : Joint/projection -> JBI (Joint par Balayage Index) et JTF (Jointure par Tri Fusion)

*R3* :



- *op1* : Restriction/projection -> RBS (Restriction par Balayage Séquentiel), pas d'index sur l'attribut *classe*.
- *op2* : Restriction/projection -> RBS
- *op3* : Joint/projection -> JBI (Joint par Balayage Index) et JTF (Jointure par Tri Fusion)
- *op4* : Joint/projection -> JBI (Joint par Balayage Index) et JTF (Jointure par Tri Fusion)

### 3. Application numérique pour la requête R1

*NT* : nombres de tuples de la relation.

*NP* : nombre de page nécessaires.

Calculez le coût des différents plans d'exécution en considérant que :

- la taille des attributs *n\_ecole*, *n\_enfant* et *n\_aliment* est de *20Ko*, la taille de tous les autres attributs est de *40Ko*.
- la taille d'une page est de *1000Ko*
- la relation *ENFANT* possède *5000* tuples
- la relation *ALLERGIE* possède *15000* tuples
- 15 des élèves sont au CP
- 20 des allergies ont comme symptômes de l'eczéma et 5 des allergies ont comme traitement la cortisone.

Restriction par balayage séquentiel (RBS) sur la relation *S* :

- $Coût = NT(S) \times CPU + NP(S)$

Jointure par boucle imbriquée (JBI) sur la relation *S* et la relation *Q* (si  $NP(S) < NP(Q)$ ) :

- $Coût = (NT(S) + NT(Q)) \times CPU + NP(S) * (1 + \frac{NP(Q)}{Buf})$

Jointure par tri fusion (JTF) sur la relation  $S$  et la relation  $Q$  :

- $Coût = (NT(S) + NT(Q)) \times CPU + CoutTri(S) + coutTri(Q) + NP(S) + NP(Q)$

Tri de la relation  $U$  (déjà mentionné) :

- $CoutTri(U) = 2 \times NP(U) \times \log_{Buf}(NP(U))$

$$CPU = 0,1$$

$$Buf = 10 \text{ (nombre de buffers consacrés aux algorithmes)}$$

**1. op1** : RBS :

$$C_1 = NT(S) \times CPU + NP(S)$$

$$\Leftrightarrow C_1 = 15000 \times 0,1 + \frac{15000 \times 120}{1000} = 3300$$

**2. op2** : RBS :

$$C_2 = NT(S) \times CPU + NP(S)$$

$$\Leftrightarrow C_2 = 5000 \times 0,1 + \frac{5000 \times 200}{1000} = 1500$$

**3. op3** : JBI :

$$C_3 = (NT(S) + NT(Q)) \times CPU + NP(S) * (1 + \frac{NP(Q)}{Buf})$$

$$\text{Soit } 20\% \times 5\% \times 15000 = 150 < 750 = 15\% \times 5000 \quad NP(S) = 150$$

$$\Leftrightarrow C_3 = (150 + 750) \times 0,1 + \frac{150 \times 20}{1000} \times (1 + \frac{750 \times 60}{10000})$$

$$\Leftrightarrow C_3 = (900) \times 0,1 + 3 \times (1 + 4,5)$$

$$\Leftrightarrow C_3 = 106,5$$

**Cout : 4906,5**

**3. op3** : JTF :

$$C_4 = (NT(S) + NT(Q)) \times CPU + CoutTri(S) + coutTri(Q) + NP(S) + NP(Q)$$

$$C_{Tri}(S) = 2 \times NP(S) \times \log_{Buf}(NP(S))$$

$$\Leftrightarrow C_{Tri}(S) = 2 \times 3 \times \log_{10}(3)$$

$$C_{Tri}(Q) = 2 \times NP(Q) \times \log_{Buf}(NP(Q))$$

Q est déjà trié sur n\_enfant.

$$\Rightarrow C_{Tri}(Q) = 0$$

$$\Leftrightarrow C_4 = (750 + 150) \times 0,1 + 2 \times 3 \times \log_{10}(3) + 3 + 45$$

$$\Leftrightarrow C_4 = 140,86$$

**Cout : 4940,86**

Imaginons que nous faisons un autre arbre :

**1. op1** : JBI :

$$C_3 = (NT(S) + NT(Q)) \times CPU + NP(S) * (1 + \frac{NP(Q)}{Buf})$$

$$\Leftrightarrow C_3 = (5000 + 15000) \times 0,1 + \frac{5000 \times 200}{1000} \times (1 + \frac{15000 \times 120}{10000})$$

$$\Leftrightarrow C_3 = 183000$$

**2. op2** : RBS :

$$C_1 = NT(S) \times CPU + NP(S)$$

$$\Leftrightarrow C_1 = 15000 \times 0,1 + \frac{15000 \times 160}{1000} = 3900$$

**Cout : 186900**

OU

**1. op1** : JTF :

$$C_4 = (NT(S) + NT(Q)) \times CPU + CoutTri(S) + coutTri(Q) + NP(S) + NP(Q)$$

$$C_{Tri}(S) = 2 \times NP(S) \times log_{Buf}(NP(S))$$

$$\Leftrightarrow C_{Tri}(S) : \text{Déjà trié sur n\_enfant} \Rightarrow 0$$

$$C_{Tri}(Q) = 2 \times NP(Q) \times log_{Buf}(NP(Q))$$

$$C_{Tri}(Q) = 2 \times 1800 \times log_{10}(1800)$$

$$\Rightarrow C_{Tri}(Q) = 0$$

$$\Leftrightarrow C_4 = (20000) \times 0,1 + 2 \times 1800 \times log_{10}(1800) + 1000 + 1800$$

$$\Leftrightarrow C_4 = 16519$$

**2. op2** : RBS :

$$C_1 = NT(S) \times CPU + NP(S)$$

$$\Leftrightarrow C_1 = 15000 \times 0,1 + \frac{15000 \times 160}{1000} = 3900$$

**Cout : 20419**