

SGBD : BASES DE DONNÉES AVANCÉES [M3106C]

TD N°1 - OPTIMISATION DE REQUÊTES

OBJECTIFS

- chemins d'accès : Index B^+
- Espace de Recherche et Optimisation
- Opérateurs de base de Jointure et Jointures assistées
- Coût d'accès et Performances

ENONCÉS

Exercice I :

On considère l'arbre B^+ d'ordre 1 de la figure 1 :

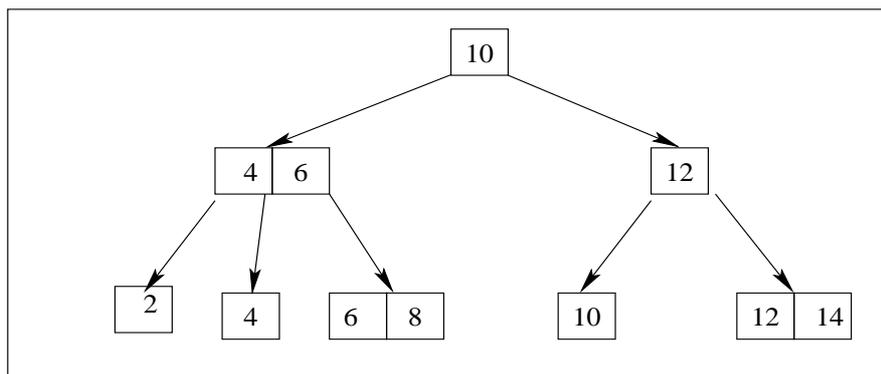


FIGURE 1. Arbre B^+ d'ordre 1

Question 1.1. Donner l'arbre B^+ obtenu après la suppression de la clé 10 de l'arbre de la figure 1.

Question 1.2. Donner l'arbre B^+ obtenu après l'insertion de la clé 9 dans l'arbre de la figure 1.

Question 1.3. Décrire une suite d'insertions de clés qui permet d'obtenir l'arbre de la figure 1.

Date: 20 mars 2014.

Hocine ABIR - IUT Villetaneuse .

On suppose que l'arbre B^+ d'ordre 1 de la figure 1 est un index sur la relation :

$Etudiant(et_numero, et_nom, et_prenom)$

dont l'instance est décrite par la figure 2. On considère la requête

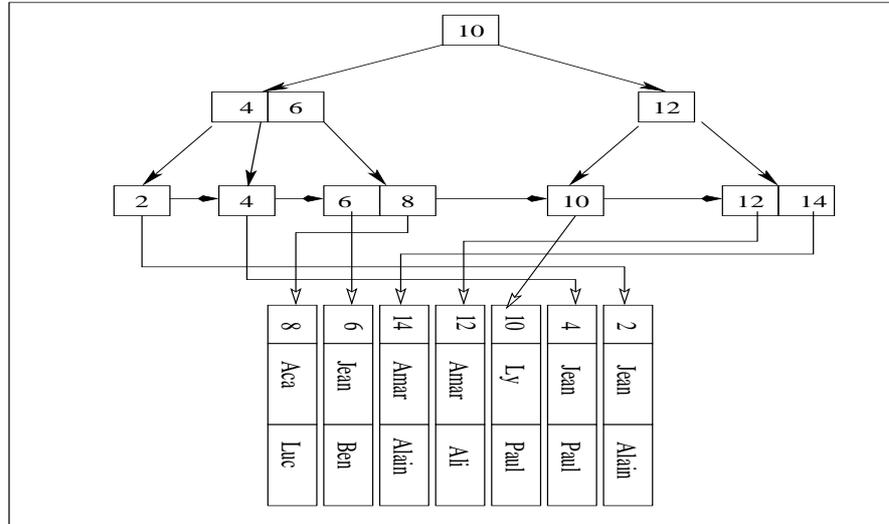


FIGURE 2. Relation $Etudiant$ et son index B^+

R suivante :

```
-----
SELECT *
FROM   Etudiant
WHERE  et_numero ≥ x
-----
```

On souhaite évaluer le coût de R en fonction de x où $x \in \{2, 4, 6, 8, 10, 12, 14\}$. Pour cela on définit le coût de R comme le nombre de tuples lus durant l'exécution de R auquel on ajoute le nombre de noeuds de l'index B^+ lus durant l'exécution de R si l'index est utilisé.

Question 1.4. Dans le cas où l'index est utilisé par R , déterminer les valeurs de x pour lesquelles le coût de R est :

- (1) le plus faible
- (2) le plus élevé

Question 1.5. Déduire les valeurs x pour lesquelles l'utilisation de l'index est préférable à l'accès séquentiel.

Exercice II :

On considère une base de données constituée de 2 relations :
R1(a,b) et R2(c,d).

- R1.b est une clé primaire.
- R2.c est une clé étrangère qui fait référence à R1.b.
- R1 a 1000000 tuples de taille 80 octets chacun,
un bloc disque peut contenir 50 tuples de R1.
- R2 a 4000 tuples de taille 200 octets chacun,
un bloc disque peut contenir 20 tuples de R2.
- L'index B^+ de R1 est d'ordre $m = 2$ sur la clé R1.b.

On souhaite évaluer le coût de la requête suivante :

```
SELECT R1.a, R1.b, R2.d
FROM R1, R2
WHERE R1.b = R2.c
```

Question 2.1. *En utilisant la jointure par boucles imbriquées (block nested loops join),*

*Déterminer le nombre minimum d'accès disque (**blocs lus**) nécessaires pour évaluer cette requête.*

Question 2.2. *En utilisant la jointure par boucles imbriquées avec index (index nested loops join),*

*Déterminer le nombre minimum d'accès disque (**blocs lus**) nécessaires pour évaluer cette requête.*

Rappel :

Le nombre minimum de niveaux d'un index B^+ d'ordre m ayant n clés est :

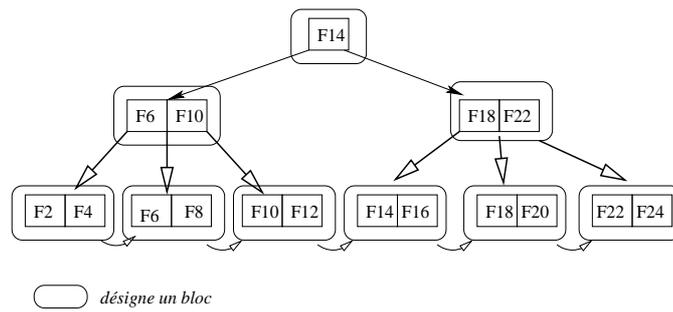
$$h^{(m,n)} = 1 + \log^{(2m+1)}\left(\frac{n}{2m}\right)$$

Exercice III :

On suppose que l'arbre B^+ de la figure 3 est un index sur la table *fournisseur(fo_id,fo_nom,fo_ville)* associé au fichier *fournisseur* décrit par la figure 4.

On s'intéresse à la requête suivante :

```
SELECT fo_nom, fo_ville
FROM fournisseur f, commande c
WHERE c.fo_num=f.fo_num
AND c.quantite > 100;
```

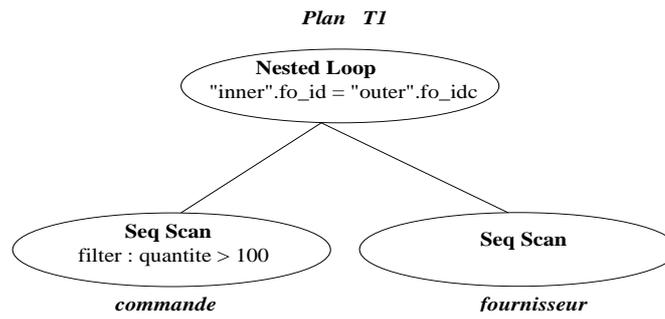
FIGURE 3. Arbre B^+ d'ordre 2

Fichier fournisseur	Fichier Commande
F2 Martin Marseille F4 Dupont Lyon	F2 200 Octobre 2007 F4 80 Janvier 2007
F6 Dubois Paris F8 Durant Marseille	F8 10 Octobre 2007 F10 30 Janvier 2007
F10 Dupuis Lyon F12 Leboeuf Lille	F6 300 Fevrier 2007 F14 45 Mars 2007
F14 Lamare Paris F16 Marvin Lyon	F4 70 Mars 2007 F10 60 Fevrier 2007
F18 Legrand Grenoble F20 Dubois Lille	F2 120 Fevrier 2007 F16 80 Janvier 2007
F22 Maury Paris F24 Nestor Marseille	F2 50 Avril 2007 F8 20 Avril 2007
	F6 40 Mai 2007 F14 10 Avril 2007
	F4 30 Mai 2007 F6 60 Juin 2007
	F2 400 Juin 2007 F16 40 Juin 2007
	F16 30 Juillet 2007 F12 60 Octobre 2007

FIGURE 4. Fichiers *fournisseur* et *commande*

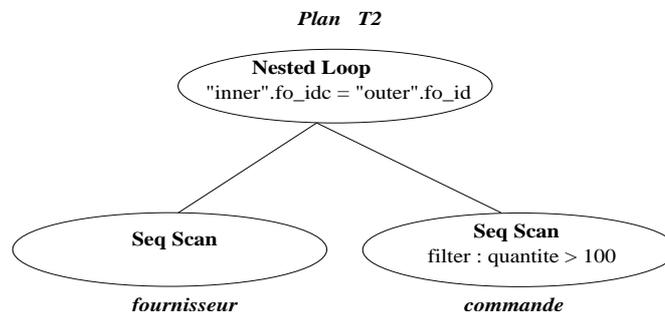
où la table $commande(fo_id, quantite, date_com)$ est associé au fichier *commande* décrit par la figure 4.

Question 3.1. Exécuter le plan $T1$ ci-dessous :



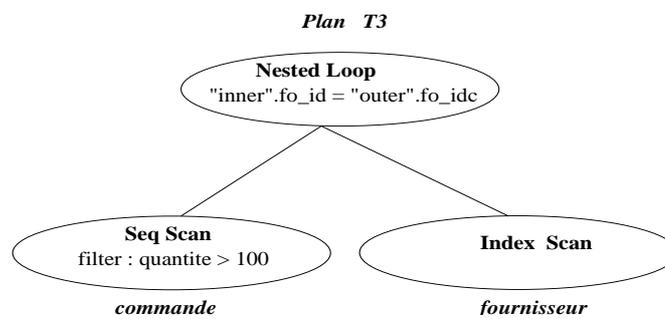
Donner le résultat obtenu (dans l'ordre) et déduire le nombre de blocs lus.

Question 3.2. Exécuter le plan T2 ci-dessous :



Donner le résultat obtenu (dans l'ordre) et déduire le nombre de blocs lus.

Question 3.3. Exécuter le plan T3 ci-dessous :



Donner le résultat obtenu (dans l'ordre) et déduire le nombre de blocs lus.