

Sidi Bel Abbès le 24 /01/2018
**Examen EMD2 : Structure Fichiers et
Structures de Données.**



Questions (6 pts)

1. Quelle est l'inconvénient principal de la méthode d'index ?
2. Pourquoi on choisira un ordre impair ($N = 2d+1$) dans un B-Arbre d'ordre N ?
3. Quelle est le rôle de l'éclatement dans un **B-Arbre**?
4. Donnez l'algorithme de recherche/insertion –Essai Linéaire-
5. Donnez les méthodes indirectes pour résoudre les collisions dans les méthodes de hachage.
6. Dans quelle phase se trouve le model entité-association.

Exercice 1 : (8 pts)

1. Ecrire un algorithme qui permet **d'afficher** les enregistrements d'un fichier (enregistrements de format fixe) en utilisant un index primaire **supérieur à un certain seuil donné**. (la clé de type entier)
2. Utiliser un parcours récursif pour ranger **les adresses des feuilles** d'un arbre de recherche binaire de gauche à droite dans une liste linéaire chaînée.
3. Charger la liste linéaire dans un fichier (Tableau non ordonné fixe).

Exercice 2 : (6 pts)

Soit à créer un fichier produit et le stocker sur disque avec une organisation aléatoire. Ce fichier contiendra les enregistrements dont les clés sont: **8, 10, 13, 18, 21, 22, 24, 25, 26** et **35**. on suppose que le fichier sera stocké sur un espace composé de **14** pages de 3 places chacune. Les **14** pages sont réparties en **9** pages principales et **5** pages de débordement.

1. Quelle sera la valeur de N Dans la fonction de hachage suivante: $f(\text{clé}) = \text{clé modulo } N$, dans les deux cas de l'application de la fonction pour la zone primaire et la zone de débordement.
2. Représenter les enregistrements du fichier en utilisant cette fonction.
3. Insérer les enregistrements **27, 36, 44, 45**.

Bon courage.

REPONSES (6 pts)

- 1- Les mises à jour sont coûteuses.
- 2- Pour simplifier la représentation.
- 3- Pour garder les nœuds équilibrés (sont remplis au minimum 50%)
- 4-

Essai linéaire : Algorithme de Recherche/Insertion

```
L1.      [Hacher]  i := h(K) { 0 <= i < M }
L2.      [Comparer]
          Si T(i) = K,
              l'algorithme se termine avec succès.
          Autrement Si T(i) est vide aller à L4.
L3.      [Avancer au prochain]
          i := i - 1
          Si i < 0 : i := i + M  Aller à L2.
L4.      [Insérer] {recherche est sans succès}
          Si N = M - 1
              débordement
          Sinon
              N := N + 1; Marquer T(i) occupé; Donnée(i) := K
```

- 5- les méthodes indirectes sont les méthodes par chaînage –interne et séparée.
- 6- La phase de conception.

Exercice 1: (8 pts)

1. **1.** algorithme qui permet **d'afficher** le contenu d'un fichier (enregistrements de format fixe) en utilisant un index primaire **supérieur à un certain seuil donne**.

TYPE T = STRUCTURE Clé : Typeclé Adr : Typeadresse FIN	TYPE Typedresse = STRUCTURE Numbloc : ENTIER Depl : ENTIER FIN	TYPE Typearticle = STRUCTURE Clé : Typeclé Info : Typeqq FIN
---	---	---

Debut

```
ouvrir( F, « FDONNEE.dat », « A »);
k:=1 ;
TQ ( k<=N)
  c=Tprim(k).cle
  Si c >seuil
    Lirebloc(F, Buf,Tprim(k).ADR.Numbloc) ;
    Afficher (Buf.Tab[Tprim(k).ADR.Depl].Article) ;
  fsi

  k:=k+1 ;
FTQ ;

Fermer(F) ;
FIN.
```

- un parcours récursif pour ranger les **adresses des feuilles** d'un arbre de recherche binaire de gauche à droite dans une liste linéaire chaînée.

```

A UN ARB ;
In UNE ACTION ;
L UNE LISTE DE ARB ;
Prec , Q DES POINTEURS VERS UNE LISTE DE ARB ;

```

```

ACTION In ( A )
  SOIT
    A UN ARB ;

  DEBUT
    SI A <> NIL
      APPEL In ( FG ( A ) ) ;

      SI ( FG ( A ) = NIL ) ET ( FD ( A ) = NIL )
        ALLOUER ( Q ) ;
        AFF_VAL ( Q , A ) ;
        AFF_ADR ( Q , NIL ) ;
        SI L = NIL :
          L := Q
        SINON
          AFF_ADR ( Prec , Q )
        FSI ;
        Prec := Q ;

      FSI ;

      APPEL In ( FD ( A ) )
    FSI
  FIN

```

- Charger la liste linéaire dans un fichier (Tableau non ordonné fixe).

```

Début
  Ouvrir(f,nomf,'n');
  P:=L; i:=1; j:=1
  TQ P<> NIL
    Si j<=b
      Buf.tab[j].cle:=p^INFO
      J:=J+1
    SINON
      buf.NB = j-1;
      EcrireDir( F, i, buf );
      buf.tab[1] ← p^INFO ;
      i ← i+1;
      j ← 2;
    FSI
  p:=p^.suiv;

FP
buf.NB ← j-1;
EcrireDir( F, i, buf );
// mettre à jour l'entête (le num du dernier bloc et
Aff-entete( F, 1, i );
Fermer( F )

```

Exercice 2 : (6pts)

- **La valeur de N** : pour le 1 cas: $N=9$
Pour le 2 cas: $N=5$
- **Représentation les enregistrements**

Exemple : $f(13) = 13 \text{ module } 9 = 4$

0	18		
1	10		
2			
3	21		
4	13	22	
5			
6	24		
7	25		
8	8	26	35

- **Insertion les enregistrements : 27, 36, 44,45.**

$f(27) = 27 \text{ module } 9 = 0$

$f(36) = 36 \text{ module } 9 = 0$

On remarque que les pages numéros 0 et 8 sont saturées. Les derniers articles (44 et 45) seront écrits dans la zone de débordement.

$f(44) = 44 \text{ module } 5 = 4$

$f(45) = 45 \text{ module } 5 = 0$

0	18	27	36	→	0	45		
1	10				1			
2					2			
3	21				3			
4	13	22			4	44		
5								
6	24							
7	25							
8	8	26	35	↘				