

Jeudi 09 Mars 2023
Professeur Khalid MEKOUAR

TP d'Algorithmique (A rendre surplace)

2^{ème} Année ESISA 2022-2023 Filière Ingénierie Logicielle

Définition

Hash coding est une technique d'organisation en utilisant un rangement dispersé direct et d'accéder à l'information en utilisant l'adressage calculé.

L'adressage calculé est assuré à l'aide d'une fonction Hashcode « F » qui à chaque identifiant « Ident » nous donne l'indice d'entrée dans le tableau THash.

Pour la fonction Hashcode on pourra prendre la fonction suivante :

- $I=L$
- 1) $F1(Ident) = \left(\sum_{i=1}^{I=L} (\text{Ord}(Ident[i]) - \text{Ord}('A')) \right) \bmod m$, avec $L=\text{long}(Ident)$.
2/3 du temps un Caractère entre 'A' et 'Z' et 1/3 du temps un Caractère entre '0' et '9'
- 2) $F2(Ident) = \left(\sum_{i=1}^{I=L} (\text{Ord}(Ident[i]) - \text{Ord}('A'))^2 \right) \bmod m$ avec $L=\text{long}(Ident)$
2/3 du temps un Caractère entre 'A' et 'Z' et 1/3 du temps un Caractère entre '0' et '9'
- 3) A trouver
- 4) A trouver

La qualité de la fonction Hashcode réside dans l'homogénéité des effectifs associés à chaque case du tableau THash.

Les sous-tables sont les différentes sous-listes associées à chaque case de THash.

TTHash = Tableau[m] de Eliste

Eliste = Structure

NbElem : Entier

PL : Pliste ;

Fin

Pliste = ^Liste

Liste = Structure

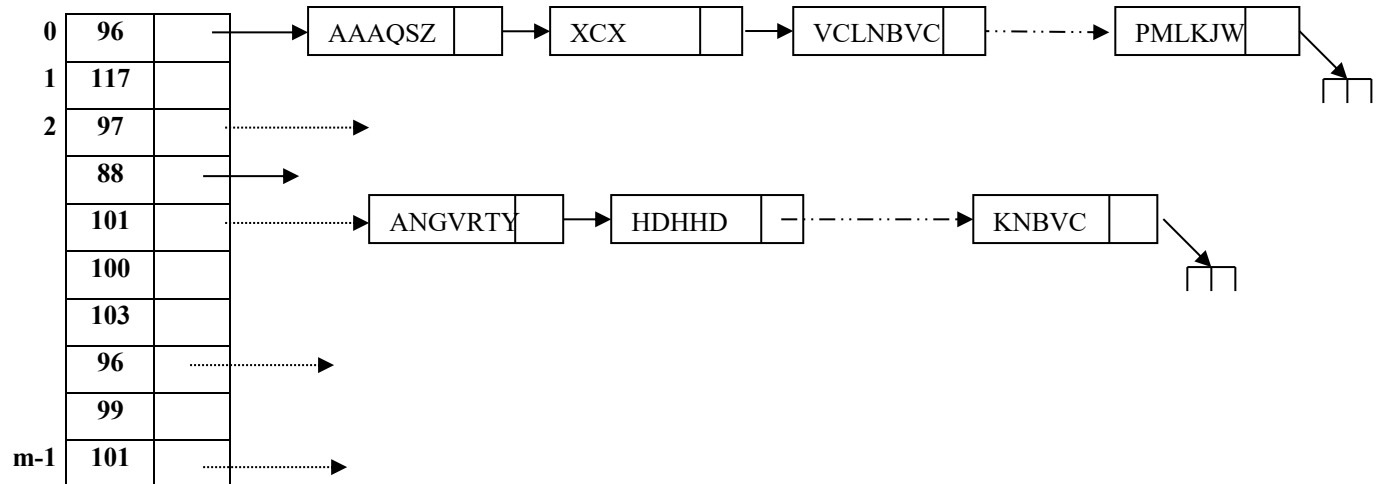
Info : Ch ;

Suiv : Pliste

Fin

m est une constante qu'on peut prendre $m=100$;

THash



Représentation

Pour simuler les deux fonctions proposées on va procéder comme suit :

- 1) Créer un fichier texte qui contient des identifiants d'une base de données ; Chaque identifiant à une longueur qui varie entre de 2 et 30.

Un identifiant par ligne

Le fichier sera créé d'une façon aléatoire de 10000 identifiants.

Chaque identifiant est de longueur variable en utilisant la fonction $\text{Random}(29)+2$ qui nous donne une valeur entre 2 et 30

Le contenu de l'identifiant est aussi constitué de caractères au hasard de 'A' à 'Z' et de «0' à '9'.

Exemple du fichier à créer

```
AA4AQ9SZ
XC1X
A3NG5VR7TY
P3MLK9JW
KJFD2JFJFF4FF8FEZ
VC3LNB7VC
KN4BV4C
HD2HH1D
MP5OLOIK7JH0YTRE6ESE
```

Procedure Crer_fichier(Name : Ch) ;

Début

.

Fin ;

Procedure Creer_Thash (Var Thash : TThash ; Name :Ch) ;

Var Ligne : CH ;

Début

;;;

Tantque non EOF(F) Faire

Lire(F, Ligne)

Ajouter_Thash(Thash, Ligne)

fintque

;

Fin ;

Procedure Ajouter Thash (Var Thash : TThash; Ligne :CH) ;

Ver Indice :Entier ;

Début

Indice = Fonc_Hashcode(Ligne, 1)

Ajouter_liste(Thash[indice].PL , Ligne)

Thash[indice].NbElem = Thash[indice].NbElem +1

Fin ;

Fonction Fonc_Hashcode(Ligne : CH ; TypeF : Entier) : Entier ;

Var i, L, Cp : Entier ;

Début

L=long(Ligne);

Case TypeF

1 : Début

Cp =0 ;

Pour i= 1 à L Faire

Si (Ligne[i]>='0') et (Ligne[i]<='P')

Alors Cp = Cp + (Ord(Ligne[i])- Ord('0'))

Sinon Cp = Cp + (Ord(Ligne[i])- Ord('A'))

FSi ;

FPour

Fonc_Hashcode = Cp Mod m ;

Fin ;

2 : Début

.

Fin ;

3 : Début

.

Fin ;

4 : Début

Fin Case

Fin ;

Procédure Ajouter_liste(Var Ptr : Pliste ; Ligne : CH) ;

Début

.

.

Fin ;

Procedure Afficher_NbElem(Var Thash : TThash) ;

Début

Pour i=0 à m-1 fire

Ecrire (Thash[i].NbElem

Fin ;

Exemple d'affichage du contenu du Tableau Thash[i].NbElem ; $0 \leq i \leq m-1$

96	102	87	103	105	99	96	95	79	112	107	100	98	98	89	104	100	91	97	91
94	108	97	98	97	100	100	101	108	97	108	106	102	91	88	88	91	101	108	108
97	91	98	91	99	98	98	86	108	107	113	98	100	100	102	91	98	99	97	98
108	113	94	108	97	105	105	105	105	84	99	103	103	100	94	93	101	94	100	101
101	100	104	98	98	91	105	100	113	94	107	91	108	100	101	101	100	101	100	98

La somme des valeurs affichées constitue les 10000 éléments du fichier initial.