

## TD de Algorithmique n° 9 : Dictionnaires

8 Décembre 2011

(Correction)

Un dictionnaire est une structure de données qui stocke des *valeurs* identifiées par des *clés*.

Les opérations élémentaires associées sont l'ajout, la suppression et la consultation de la valeur associée à une clé.

### I) Listes d'association

La première idée est de stocker un dictionnaire sous forme d'une liste de paires (clé, valeur).

#### Exercice 1 :

1. Comment implémenter-t-on dans ce cas les opérations élémentaires ?
2. Quel est le coût de l'opération dans le pire cas ?

#### Correction :

1. 

```
let add a b l = (a,b) :: l
let rec remove a = function
  | [] -> []
  |(ta,_) :: q when ta=a -> q
  |t :: q -> t::(remove a q)
let find a = function
  | [] -> raise Not_found
  |(ta,tb) :: q when a = ta -> tb
  |_ :: q -> find a q
```
2. constant, linéaire et linéaire

### II) Fonctions de hachage

Pour accélérer la procédure, il est possible d'utiliser des arbres si l'on sait trier les clés. Il est aussi possible de tenter de ramener les clés à des entiers de  $[1..n]$  puis d'utiliser des tableaux. La fonction qui transforme les clés est appelée *fonction de hachage*.

#### Exercice 2 :

À partir des exemples de l'exercice suivant, tentez de trouver les propriétés que doit avoir une telle fonction.

#### Correction :

- déterministe
- a valeur dans  $[1..n]$
- peu de collision
- calcul efficace

**Exercice 3 : Les entiers**

On veut placer des entiers dans un tableau de longueur  $n$  (dont les cases sont numérotées entre 0 et  $n - 1$ ), en utilisant une fonction de hachage. On suppose que les valeurs à placer ont été tirées aléatoirement de façon uniforme, c'est-à-dire que toutes les valeurs ont la même probabilité d'apparition. Expliquer pourquoi les fonctions suivantes sont ou non des bonnes fonctions de hachage. L'entier  $k$  désigne l'élément considéré :

1.  $h(k) = 1$ ,
2.  $h(k) = \lfloor k/n \rfloor$ ,
3.  $h(k) = k + \text{RAND}(n) \text{ MOD } n$ , où  $\text{RAND}(n)$  est une fonction qui renvoie un entier aléatoire entre 0 et  $n - 1$ ,
4.  $h(k) = k \text{ MOD } n$

**Correction :**

1. La fonction constante 1 est une fonction de hachage très mauvaise parce qu'elle engendre trop de collisions.
2. La fonction de hachage doit être une fonction  $h : U \rightarrow \{0, \dots, n-1\}$ . Ici, si  $k \gg n$  on a  $h(k) > n-1$ , donc cette fonction ne peut pas être utilisée comme fonction de hachage.
3. Cette fonction n'est pas une bonne fonction de hachage parce qu'elle n'est pas déterministe.
4. Cette fonction est une bonne fonction de hachage, appelée "la méthode de division".

**Exercice 4 : Les chaînes de caractères**

On propose la fonction de hachage suivante sur les chaînes de caractère de longueur  $\ell$ , à valeurs dans l'intervalle  $[0, p]$ , où  $p$  est un nombre premier :

$$h(s) = \sum_{i=0}^{\ell-1} s[i]b^{\ell-1-i} \text{ MOD } p,$$

où  $b$  est une puissance de 2 et  $s[0]$  le caractère plus à gauche de la chaîne  $s$ .

1. Pour  $\ell = 4$ ,  $p = 7$ , et  $b = 8$ , calculez les valeurs de hachage associées aux valeurs des chaînes "info" et "bbio" (en sachant qu'on a les codes ascii suivants :  $\text{ascii}[i] = 105$ ,  $\text{ascii}[n] = 110$ ,  $\text{ascii}[f] = 102$ ,  $\text{ascii}[o] = 111$ ,  $\text{ascii}[b] = 98$ ),
2. Expliquez l'intérêt (du point de vue du programmeur) de prendre  $b$  comme puissance de 2.
3. Recommencez le calcul de la première question avec  $p = 4$ . Pourquoi choisit-on  $p$  premier ?

**Correction :**

1. Dans ce cas on a  $h(s) = (s[0]8^3 + s[1]8^2 + s[2]8^1 + s[3]8^0) \text{ MOD } 7$ .  
 $h(\text{"info"}) = (105 \cdot 512 + 110 \cdot 64 + 102 \cdot 8 + 111 \cdot 1) \text{ MOD } 7 =$   
 $(105 \cdot 512 + 110 \cdot 64 + 102 \cdot 8 + 111 \cdot 1) \text{ MOD } 7 = 1,$   
 $h(\text{"bbio"}) = (98 \cdot 512 + 98 \cdot 64 + 105 \cdot 8 + 111 \cdot 1) \text{ MOD } 7 =$   
 $(98 \cdot 512 + 98 \cdot 64 + 105 \cdot 8 + 111 \cdot 1) \text{ MOD } 7 = 6.$
2. Parce que multiplier par 2 c'est facile.
3.  $(105 \cdot 512 + 110 \cdot 64 + 102 \cdot 8 + 111 \cdot 1) \text{ MOD } 4 = 3,$   
 $(98 \cdot 512 + 98 \cdot 64 + 105 \cdot 8 + 111 \cdot 1) \text{ MOD } 4 = 3.$

### III) Tables de hachage

**Exercice 5 :**

Si l'on représente un dictionnaire sous la forme d'un tableau de listes d'association tel que les éléments de  $t[i]$  sont les éléments du dictionnaire dont la clé a pour hash  $i$ .

Comment implémente-t-on les opérations élémentaires ?