



uOttawa

CSI 2510 - Structures de données et algorithmes

AUTOMNE 2021 SECTION A

PROFESSEURE DORRA RIAHI

---

## Devoir 7

---

Kien Do (ID: 300163370)

**Question 1. [3 points]**

Une structure de données est composée de  $n$  éléments avec clé et sur laquelle des opérations sont effectuées. Quelle est la complexité au pire cas en temps d'exécution pour les opérations suivantes :

1.

	Insertion d'un élément Insert an element with a key "5".	Retrait d'un élément Delete an element with a key "4"
Tableau trié Sorted sequence (array implementation)	Recherche de l'élément $O(\quad)$ – Insertion à la position trouvée $O(\quad)$ $O(\log n)$ Search for position – $O(\quad)$ Insert at the given position – $O(\quad)$	Recherche de l'élément $O(\quad)$ – Retrait à la position trouvée $O(\quad)$ $O(\log n)$ Search for position – $O(\quad)$ Delete from the given position – $O(\quad)$
Liste chaînée simple (non-triée) Unsorted sequence (singly linked list implementation)	Recherche de l'élément $O(\quad)$ – Insertion à la position trouvée $O(\quad)$ $O(n)$ Search for position – $O(\quad)$ Insert at the given position – $O(\quad)$	Recherche de l'élément $O(\quad)$ – Retrait à la position trouvée $O(\quad)$ $O(n)$ Search for position – $O(\quad)$ Delete from the given position – $O(\quad)$

Monceau-max dans un tableau Max heap (array implementation)	Insertion $O(\quad)$ Insert – $O(\log n)$	Recherche de l'élément $O(\quad)$ – Retrait à la position trouvée $O(\quad)$ Search for position – $O(\log n)$ Delete – $O(\log n)$
AVL	Recherche de l'élément $O(\quad)$ – Insertion à la position trouvée $O(\quad)$ Search for position – $O(\log n)$ Insert at the given position – $O(\log n)$	Recherche de l'élément $O(\quad)$ – Retrait à la position trouvée $O(\quad)$ Search for position – $O(\log n)$ Delete from the given position – $O(\log n)$
Table de hachage (sans aucune collision) Hash table without any collision	Calcul de la position de l'élément $O(\quad)$ – Insertion à la position trouvée $O(\quad)$ Determine position – $O(1)$ Insert at the given position – $O(1)$	Recherche de l'élément $O(\quad)$ – Insertion à la position trouvée $O(\quad)$ Search for position – $O(1)$ Delete from given position – $O(1)$

**Question 2. [4.5 points]**

Soit la table de hachage de dimension 7 montrée ci-dessous dans laquelle les insertions se font en utilisant une fonction de hachage  $h(k) = 2*k \bmod 7$ . Insérer les clés suivantes dans cet ordre: 2, 8, 1, 15, 3.

Bien montrer les positions sondées pour chaque insertion.

Consider the following Hash table where insertions are done using the hash function  $h(k) = 2*k \bmod 7$ . Insert the following keys 2, 8, 1, 15, 3 in an empty Hash table with 7 spots and **underlining/squaring/circling the positions (underline/square/circle a key on the position)** that are probed for each insertion. Show the Hash table after each insertion.

E.g. (insertion de 1, 3 et 8 avec sondage linéaire) (insert 1, 3 and 8 with linear probing)

0	1	2	3	4	5	6
		<u>1</u>				
		1				<u>3</u>
		1	8			3

Après ces insertions, quel est le nombre moyen de sondage à effectuer afin de rechercher une clé dans cette table? Effectuer cet exercice pour les trois méthodes de résolution de collision suivantes :

After all insertions, what is the **average number of probes** (comparisons) needed to search for a key in this table? In each part collisions are solved by each of the following 3 methods.

2.

a) Sondage linéaire. Linear probing.

Table vide

0	1	2	3	4	5	6

Insérer 2

$h(2) = 4$

0	1	2	3	4	5	6
				2		

Insérer 8

$h(8) = 2$

0	1	2	3	4	5	6
		8		2		

Insérer 1

$h(1) = 2$

Probe:  $(2 + 1) \bmod 7 = 3$

0	1	2	3	4	5	6
		8	1	2		

**Insérer 15**

$$h(15) = 2 \cdot 15 \bmod 7 = 30 \bmod 7 = 2$$

$$\text{Probe: } (2 + 1) \bmod 7 = 3$$

$$\text{Probe: } (3 + 1) \bmod 7 = 4$$

$$\text{Probe: } (4 + 1) \bmod 7 = 5$$

0	1	2	3	4	5	6
		8	1	2	15	

**Insérer 3**

$$h(3) = 2 \cdot 3 \bmod 7 = 6$$

0	1	2	3	4	5	6
		8	1	2	15	3

Nombre moyen de sondages =  $(0 + 0 + 1 + 3 + 0)/5 = 0.8$  probes.

b) Sondage quadratique. Quadratic probing.

(2) [1.5 points] Sondage quadratique. Quadratic probing

0	1	2	3	4	5	6
				2		
		8		2		
		"	1	"		
		"	"	"		15
3		8	1	2		15

Nombre moyen de sondages =  $5/5 = 1$   
Average number of probes =

c) Hachage double.

(3) [2 points] Hachage double avec la fonction secondaire  $d(k) = 4 - (k \bmod 3)$ .

Double hashing with secondary hash function:  $d(k) = 4 - (k \bmod 3)$ .

Double hashing is done by  $h_j(k) = [h(k) + j \cdot d(k)] \bmod N$ , where  $N=7$ .

0	1	2	3	4	5	6
				2		
		8		"		
		"	1	"		
		"	"	"		15
3		8	1	2		15

Nombre moyen de sondages =  $4/5 = 0.8$   
Average number of probes =

**Question 3. [4.5 points]**

Soit une table de hachage construite en utilisant le hachage double

$$h_i(k) = [h(k) + i \cdot d(k)] \bmod 11$$

La fonction de hachage primaire est  $h(k) = (k \bmod 11)$  et la fonction secondaire est  $d(k) = (k \div 5) + 1$  avec div étant la division entière (e.g.  $10 \bmod 3 = 1$  and  $10 \div 3 = 3$ ).

Dans les tableaux ci-dessous, montrer l'état de la table après chaque insertion, retrait ou recherche en identifiant bien les éléments sondés.

3.

(1) [2 point]

Insérer les clés 13 et 16 dans la table suivante en montrant bien chacune des étapes.

Insert the keys 13 and 16 in the following hash table and describe the process.

i =	0	1	2	3	4	5	6	7	8	9	10
			2	3							
			//	//		13					
			//	//		13				16	

**Description:**

Insertion de la clé 13.

$$h(13) = 13 \bmod 11 = 2 \quad \leftarrow \text{déjà occupé par 2}$$

$$d(13) = (13 \div 5) + 1 = 2 + 1 = 3$$

$$(j = 1) \quad [2 + (1)(3)] \bmod 11 = 5 \quad \leftarrow \begin{array}{l} \text{disponible} \\ \text{faire l'insertion à } i = 5 \end{array}$$

Insertion de la clé 16.

$$h(16) = 16 \bmod 11 = 5 \quad \leftarrow \text{déjà occupé par 13}$$

$$d(16) = (16 \div 5) + 1 = 3 + 1 = 4$$

$$(j = 1) \quad [5 + (1)(4)] \bmod 11 = 9 \quad \leftarrow \begin{array}{l} \text{disponible} \\ \text{faire l'insertion à } i = 9 \end{array}$$

(2) [0.5 point]

Retirer les clés 2 et 3 dans la table obtenue en (1) en montrant bien chacune des étapes.

Delete the key 2 and 3 in the hash table obtained in (1).

i =	0	1	2	3	4	5	6	7	8	9	10
			D	D		13				16	

**Description:**

Retirer la clé 2 (on fait la même chose avec la clé 3).

$$h(2) = 2 \bmod 11 = 2 \quad \leftarrow \begin{array}{l} \text{clé est trouvé} \\ \text{remplace avec la clé D (disponible)} \end{array}$$

**(3) [1 point]**

Rechercher la clé 13 dans la table obtenue en (2) en montrant bien chacune des étapes.

Search the key 13 in the hash table obtained in (2) and describe the process.

i =

0	1	2	3	4	5	6	7	8	9	10
		D	D		13				16	

Rechercher la clé 13.

$$h(13) = 13 \bmod 11 = 2 \quad \leftarrow \begin{array}{l} D \neq 13 \\ 13 \neq 13 \end{array}$$

$$d(13) = (13 \div 5) + 1 = 2 + 1 = 3$$

$$(j = 1) \quad [2 + (1)(3)] \bmod 11 = 5 \quad \leftarrow \begin{array}{l} 13 == 13 \\ \text{retourne l'élément à } i = 5 \end{array}$$

**(4) [1 point]**

Insérer la clé 38 dans la table obtenue en (3) en montrant bien chacune des étapes.

Insert the key 38 in the hash table obtained in (3), and describe the process

i =

0	1	2	3	4	5	6	7	8	9	10
		38	D		13				16	

Insérer la clé 38.

$$h(38) = 38 \bmod 11 = 5 \quad \leftarrow \begin{array}{l} \text{déjà occupé par 13} \\ \text{déjà occupé par 16} \end{array}$$

$$d(38) = (38 \div 5) + 1 = 3 + 1 = 4$$

$$(j = 1) \quad [5 + (1)(4)] \bmod 11 = 9 \quad \leftarrow \begin{array}{l} \text{déjà occupé par 16} \\ j++ \end{array}$$

$$(j = 2) \quad [2 + (2)(4)] \bmod 11 = 2 \quad \leftarrow \begin{array}{l} \text{disponible} \\ \text{remplace D avec 38} \end{array}$$