Data Structures and its Applications

V R BADRI PRASAD
Department of Computer Science & Engineering

DATA STRUCTURES AND ITS APPLICATIONS

Double Hashing

V R BADRI PRASAD
Department of Computer Science & Engineering

Data Structures and its Applications

Open Hashing – Double Hashing

- Double hashing is collision resolving technique in open addressed hash tables.
- Double hashing uses the idea of applying a second hash function to key when a collision occurs.

Let **h1(key) = *key mod 13*** be hash function1.

Let ***h2(key) = 7 – key mod 7*** be hash function2.

Consider the following elements to be inserted into the hash table of size **TABLESIZE**.

Keys : { 18, 41, 22, 44 }.

Let's now calculate the values of h1(key) and h2(key).

Key	h1(key)	h2(key)
18	18 mod 13 = 5	7-(18 mod 7) = 3
41	41 mod 13 = 2	7-(41 mod 7) = 1
22	22 mod 13 = 9	7-(22 mod 7) = 6
44	44 mod 13 = 5	7-(44 mod 7) = 2

Data Structures and its Applications

Open Hashing – Double Hashing

Key	h1(key)	h2(key)	Double hash(key)
18	18 mod 13 = 5	7-(18 mod 7) = 3	-
41	41 mod 13 = 2	7-(41 mod 7) = 1	-
22	22 mod 13 = 9	7-(22 mod 7) = 6	-
44	44 mod 13 = 5	7-(44 mod 7) = 2	-

Index	0	1	2	3	4	5	6	7	8	9	10	11	12
Key			41			**18**				22			

- Key – 18 , using h1(key) gives 5 as index / hash. Go to location 5. It is Free. Assign 18 to location 5.
- Key – 41 , using h1(key) gives 2 as index / hash. Go to location 2. It is Free. Assign 41 to location 2.
- Key – 22 , using h1(key) gives 2 as index / hash. Go to location 2. It is Free. Assign 22 to location 9.
- Key – 44 , using h1(key) gives 5 as index / hash. Go to location 5. It is not Free.

Use double hashing function.

Open Hashing – Double Hashing

Key	h1(key)	h2(key)	Double hash(key)
18	18 mod 13 = 5	7-(18 mod 7) = 3	-
41	41 mod 13 = 2	7-(41 mod 7) = 1	-
22	22 mod 13 = 9	7-(22 mod 7) = 6	-
44	44 mod 13 = 5	7-(44 mod 7) = 5	-

Index	0	1	2	3	4	5	6	7	8	9	10	11	12
Key			41			18				22			

- Key – 44 , using h1(key) gives 5 as index / hash. Go to location 5. It is not Free.

Use double hashing function. Index/hash = hash1(key) + j * hash2(key) , j=1 as it has the first collision.

. Index/hash =(5 + 1 * (5)) % 13 = 10

Open Hashing – Double Hashing

Key	h1(key)	h2(key)	Double hash(key)
18	18 mod 13 = 5	7-(18 mod 7) = 3	-
41	41 mod 13 = 2	7-(41 mod 7) = 1	-
22	22 mod 13 = 9	7-(22 mod 7) = 6	-
44	44 mod 13 = 5	7-(44 mod 7) = 5	**10**

Index	0	1	2	3	4	5	6	7	8	9	10	11	12
Key			41			18				22	44		

- Key – 44 , using h1(key) gives 5 as index / hash. Go to location 5. It is not Free.

Use double hashing function. Index/hash = hash1(key) + j * hash2(key) , j=1 as it has the first collision.

Index/hash =(5 + 1 * (5)) % 13 = 10.

Since location 10 is free, key value 44 is stored in location 10.

The collision is resolved using double hashing.

THANK YOU

V R BADRI PRASAD
Department of Computer Science & Engineering
badriprasad@pes.edu