COMPUTER SCIENCE AND DA

Data Structures through Python

Queues and Hash Tables

Lecture No. 5

By- Kashif Sir

Topics to be covered

→ Double Ended Queue (Deque)

↳ Enqueue-rear (D, value)

↳ Enqueue-front (D, value)

↳ Dequeue-front (D)

↳ Dequeue-rear (D)

Agenda ⇒ Priority Queue

⇒ Hash Tables

Priority Queue

Priority Queue is a special Queue in which element is deleted on the basis of priority not on the basis of arrival sequence

delete elements from higher priority to lower priority : max heap

delete elements from lower priority to higher priority => min heap

In Priority Queue every element has two attribute i.e, value & priority

PQ = { 'A' : 2 , 'B' : 3 , 'C' : 1 , 'D' : 5 , 'E' : 4 }

higher value - higher priority

```
PQ.sort()
  for i,j in PQ
    PQ.pop()
```

Deletion sequence : 'D' 'E' 'B' 'A' 'C'

Insertion Sequence : 'A' 'B' 'C' 'D' 'E'

Consider a Deque

1	~~4~~ 10	~~7~~	9	~~6~~ 8	5	3	2
0	1	2	3	4	5	6	7

r

r = 2 , f = 4

Dequeue_front( ) r = 2, f = 5

Dequeue_rear( ) r = 1, f = 5

Enqueue_front(8) r = 1, f = 4

Enqueue_front(9) r = 1, f = 3

Dequeue_rear( ) r = 0, f = 3

Enqueue_rear(10) r = 1, f = 3

Print (r + f)

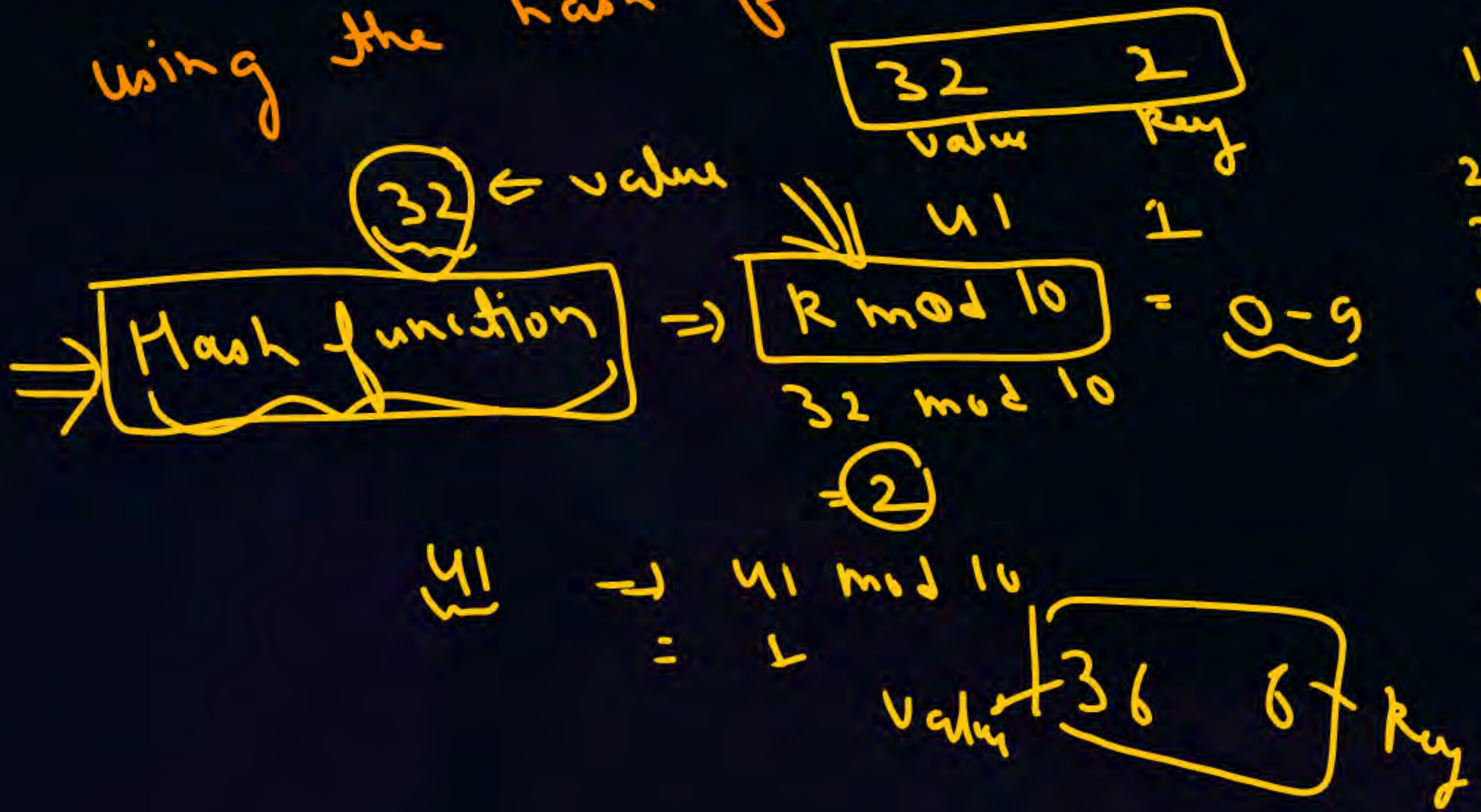
$1 + 3$

$= \boxed{4}$

Hash Table

Hash table is a linear data structure.

It is used to create mapping between the value & key using the hash function.



Hash Table

0	
1	41
2	32
3	
4	
5	
6	
7	
8	
9	

Hashing

Collision : In hash table if key of two values through the hash function is same then it creates collision.

1) Open hashing

2) Closed hashing
   - Linear Probing
   - Quadratic Probing
   - Double Hashing

Summary

1) Priority Queue

2) Hash Table

THANK - YOU