# CS & IT ENGINEERING

## Data Structure & Programming

**Hashing**

**Lec- 02**

By- Pankaj Sharma Sir

# TOPICS TO BE COVERED

- Hashing 02

free from primary clustering problem

# Quadratic Probing

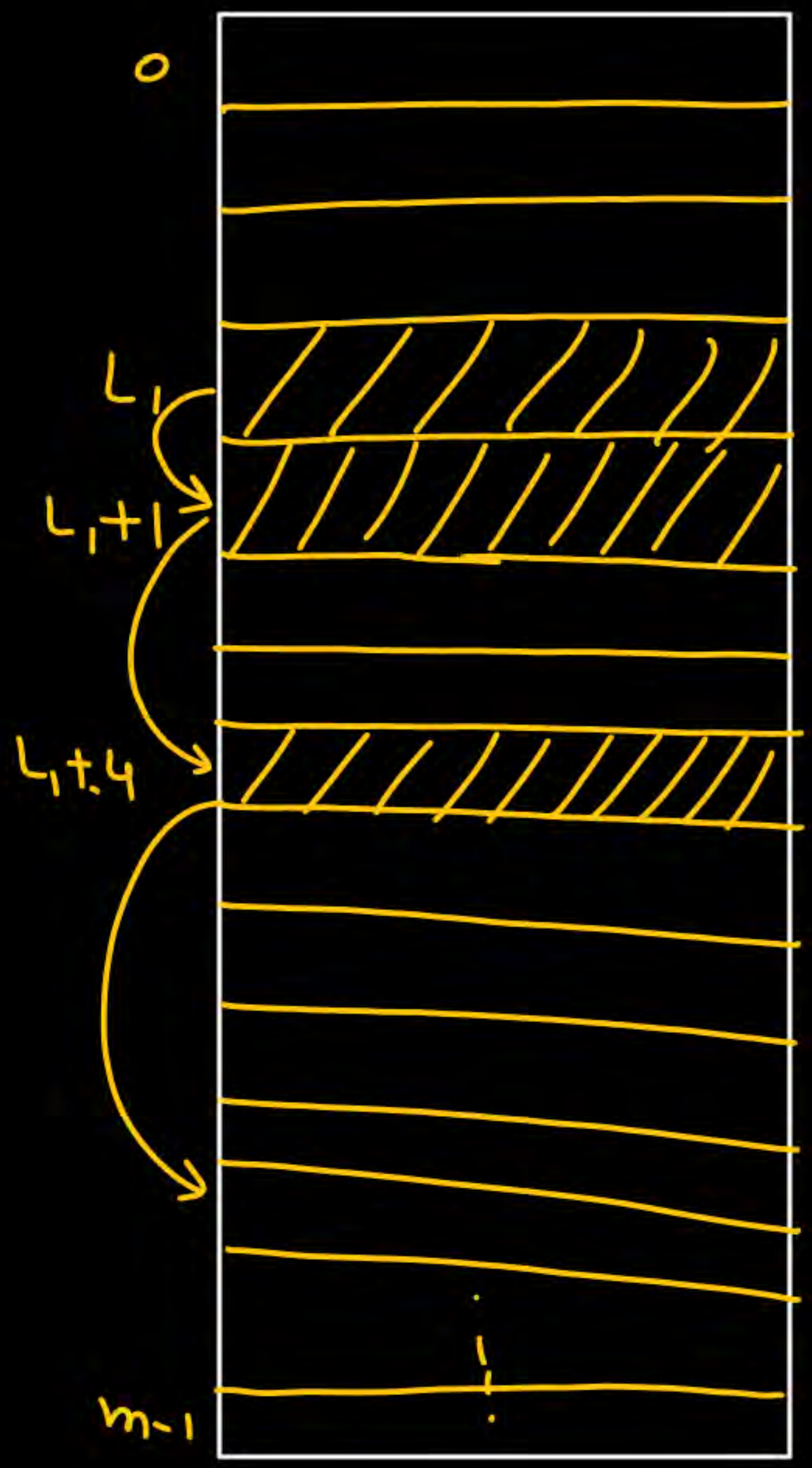
$$h(k) = k \bmod m = L_1 \Big]\ \text{collission}$$

$$H(k, i) = (h(k) + i^2) \bmod m$$

$$H(k, 1) = (h(k) + 1^2) \bmod m = L_1 + 1$$

$$H(k, 2) = (h(k) + 2^2) \bmod m = L_1 + 4$$

$$H(k, 3) = (h(k) + 3^2) \bmod m = L_1 + 9$$



Keys : 24, 17, 32, 2, 13, 50, 30, 61

$m = 11$

$h(k) = k \bmod m$

(i) $h(24) = 24 \bmod 11 = 2$

(ii) $h(17) = 17 \bmod 11 = 6$

(iii) $h(32) = 32 \bmod 11 = 10$

(iv) $h(2) = 2 \bmod 11 = 2$ collision

$H(2,1) = (h(2) + 1^2) \bmod 11$

$= 3 \bmod 11 = 3$ ✓

(v) $h(13) = 13 \bmod 11 = 2$ collission

$H(13,1) = (h(13) + 1^2) \bmod 11$

$= 3$ collission

$H(13,2) = (h(13) + 2^2) \bmod 11$

$= 6$ collission

$H(13,3) = (h(13) + 3^2) \bmod 11$

$= 0$ ✓

(vii) $h(50) = 50 \bmod 11 = 6$ collision

$H(50,1) = (h(k) + 1^2) \bmod 11 = 7$ ✓

(viii) $h(30) = 30 \bmod 11 = 8$

(ix) $h(61) = 61 \bmod 11 = 6$ collission

$H(61,1) = (h(k) + 1^2) \bmod 11 = 7$ collission

$H(61,2) = (h(k) + 2^2) \bmod 11$

$= 10$ collission

$H(61,3) = (h(k) + 3^2) \bmod 11$

$= 4$ ✓

| | |
|---|---|
| 0 | 13 |
| 1 | |
| 2 | 24 |
| 3 | 2 |
| 4 | 61 |
| 5 | |
| 6 | 17 |
| 7 | 50 |
| 8 | 30 |
| 9 | |
| 10 | 32 |

# Secondary Clustering Problem

$m = 11 \quad h(k) = k \bmod m$

$h(24) = 24 \bmod 11 = 2$

$h(2) = 2 \bmod 11 = 2$

$h(13) = 13 \bmod 11 = 2$

$i = 1$

$H(24,1) = (h(24) + 1^2) \bmod 11 = 3$

$H(2,1) = (h(2) + 1^2) \bmod 11 = 3$

$H(13,1) = (h(13) + 1^2) \bmod 11 = 3$

$i = 2$

$H(24,2) = (h(24) + 2^2) \bmod 11 = 6$

$H(2,2) = (h(2) + 2^2) \bmod 11 = 6$

$H(13,2) = (h(13) + 2^2) \bmod 11 = 6$

$i = 3$

$H(24,3) = (h(24) + 3^2) \bmod 11 = 0$

$H(2,3) = (h(2) + 3^2) \bmod 11 = 0$

$H(13,3) = (h(13) + 3^2) \bmod = 0$

$i = 4 \quad H(24,4) = (h(24) + 4^2) \bmod 11 = 7$

$H(2,4) = 7$

$H(13,4) = 7$

$i = 5 \quad H(24,5) = (h(24) + 5^2) \bmod 11 = 5$

$H(2,5) = 5$

$H(13,5) = 5$

2, 3, 6, 0, 7, 5 $\longrightarrow$

resolution path same

$i = 6$

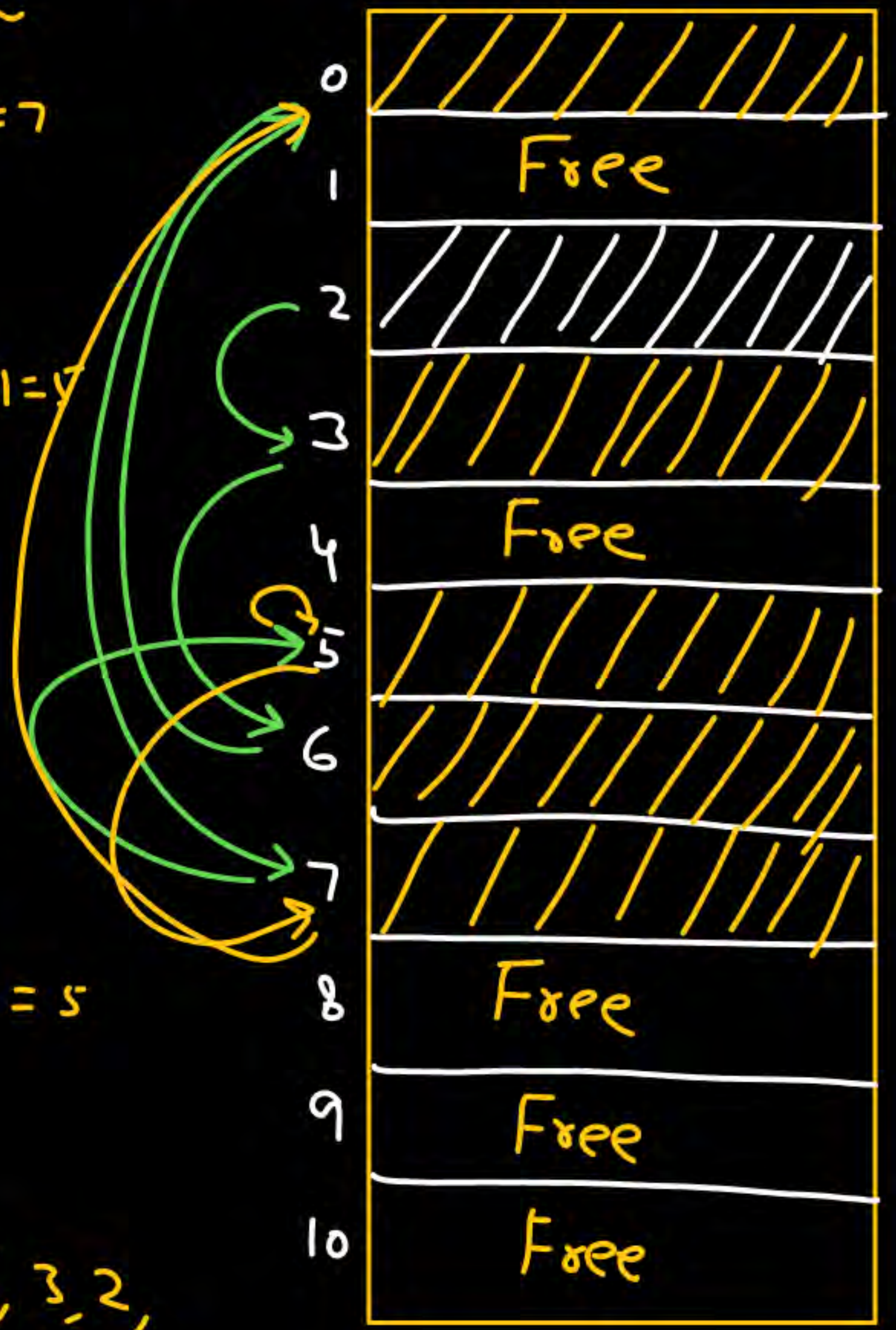
$H(24,6) = (h(24) + 6^2) \bmod 11 = 5$

$H(2,6) = 5$

$H(13,6) = 5$

$i = 7$ 2, 3, 6, 0, 7, 5, 5, 7, 0, 6, 3, 2,

$m = 11$




occupied $\frac{m+1}{2}$ = 6 slot

Keys that are hashed to same memory location always follows same resolution path bcz of which we are not able to utilize the table size Efficiently.

Inspite of almost 50% free slots, we are not able to provide a free slot to new element.

# Secondary Clustering Problem $m = 11$

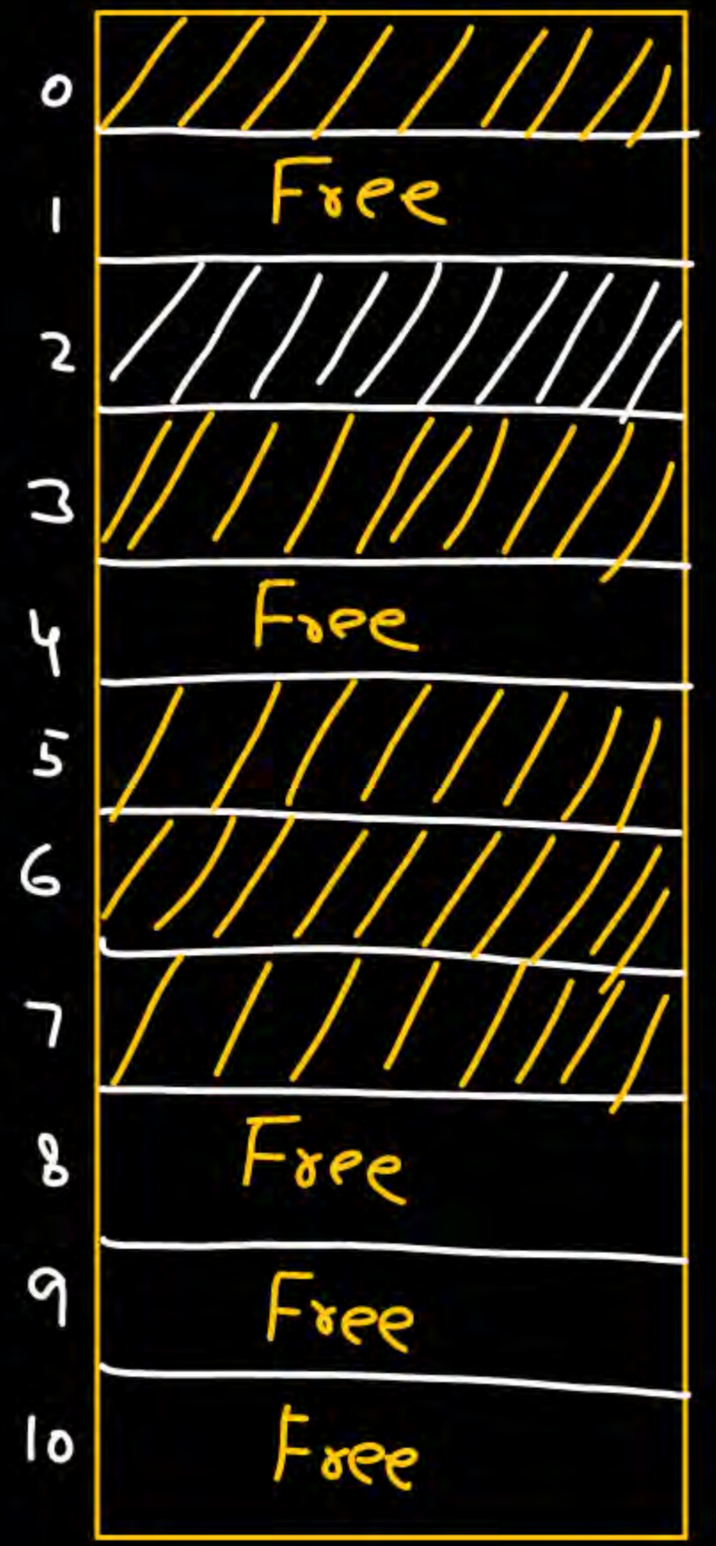
$K = 35$

$K = 46$

$K = 57$

$K = 68$

$\vdots$



# Double Hashing

Let $h(k)$ is the hash we are using

$h(k) = k \bmod m \Rightarrow$ Collission occur

Linear Probing $H(k,i) = (h(k)+i) \bmod m$

Quad. Probing $H(k,i) = (h(k)+i^2) \bmod m$

$H(k,i) = (h(k) + i\,h'(k)) \bmod m$ Double hashing

$h(k)$: Primary hash function

$h'(k)$: Secondary hash function

$$H(k,i) = (h(k) + i\,h'(k)) \bmod m$$

$$\boxed{h'(k) = k \bmod m}$$

$$h'(k) = k \bmod m + 1 \quad ✓$$

Secondary hash function

$\downarrow$

$h'(k)$ can never be 0

$$H(k,i) = (h(k) + i \times 0) \bmod m$$

$$= h(k) \bmod m$$

$\Rightarrow$ collission

Q Keys : 13, 17, 21, 2, 57, 28, 30, 27

$h(x) = x \bmod 11 \Rightarrow \boxed{m = 11}$

$h'(x) = 7 - (x \bmod 7)$

(i) $h(13) = 13 \bmod 11 = 2$ ✓

(ii) $h(17) = 17 \bmod 11 = 6$ ✓

(iii) $h(21) = 21 \bmod 11 = 10$ ✓

(iv) $h(2) = 2 \bmod 11 = 2$

Collission

$H(2,1) = (h(k) + 1 \cdot h'(k)) \bmod 11$

$h'(2) = 7 - 2 \bmod 7 = 5$

$H(2,1) = (2 + 1 \cdot 5) \bmod 11 = 7$ ✓

v) $h(57) = 57 \bmod 11 = 2$

Collission

$H(57,1) = (h(57) + 1 \cdot h'(57)) \bmod 11$

$= (2 + 6) \bmod 11 = 8$ ✓

vi) $h(28) = 28 \bmod 11 = 6$

Collission

$H(28,1) = (h(28) + 1 \cdot h'(28)) \bmod 11$

$= (6 + 7) \bmod 11 = 2$

Collission

$H(28,2) = (h(28) + 2 \cdot h'(28)) \bmod 11$

$= (6 + 2 \cdot 7) \bmod 11 = 9$ ✓

vii) $h(30) = 30 \bmod 11 = 8$

Collission

$H(30,1) = (h(30) + 1 \cdot h'(30)) \bmod 11$

$= (8 + 5) \bmod 11 = 2$

Collission

$H(30,2) = (h(30) + 2 \cdot h'(30)) \bmod 11$

$= (8 + 2 \cdot 5) \bmod 11 = 7$

Collission

$H(30,3) = (h(30) + 3 \cdot h'(30)) \bmod 11$

$= (8 + 3 \cdot 5) \bmod 11 = 1$ ✓

viii) $h(27) = 27 \bmod 11 = 5$ ✓

$h(2) = 2, 7$

$h(57) = 2, 8$

$H(2,1) = (h(2) + h'(2)) \bmod 11$

$= (2 + 5) \bmod 11 = 7$

$H(57,1) = (h(57) + h'(57)) \bmod 11$

$= (2 + 6) \bmod 11 = 8$

Overhead
↳ computing 2 hash function

more Time complexity

## Load factor ($\lambda$)

$$\lambda = \frac{n}{m}$$

$n \rightarrow$ no. of keys

$m \rightarrow$ table size

| | | |
|---|---|---|
| $n = 20$ | $n = 30$ | $n = 60$ |
| $m = 40$ | $m = 40$ | $m = 40$ |
| $\lambda = \frac{20}{40} = \frac{1}{2}$ | $\lambda = \frac{30}{40} = \frac{3}{4}$ | $\lambda = \frac{60}{40}$ |

$\boxed{\lambda \leq 1}$

Linear Probing

Quad. Probing

Double Hashing

$\boxed{\lambda > 1}$

Separate chaining

Collission resolve

list of elem

Keys: 400, 300, 125, 625, 36, 96, 106, 500

$m = 10$

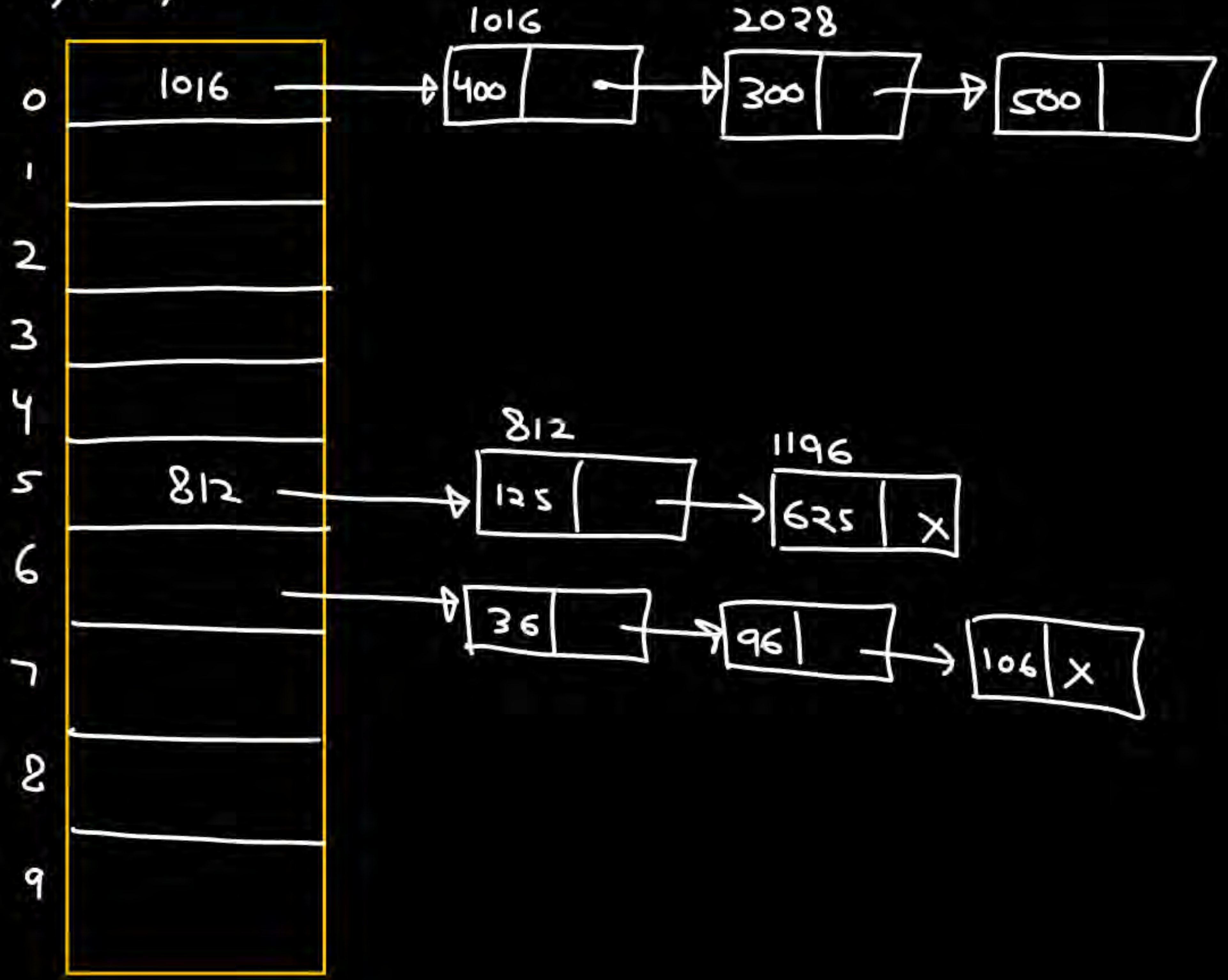
$h(400) = 400 \bmod 10 = 0$

$h(300) = 300 \bmod 10 = 0$ ] collission

$h(125) = 125 \bmod 10 = 5$

$h(625) = 625 \bmod 10 = 5$

$h(36) = 36 \bmod 10 = 6$

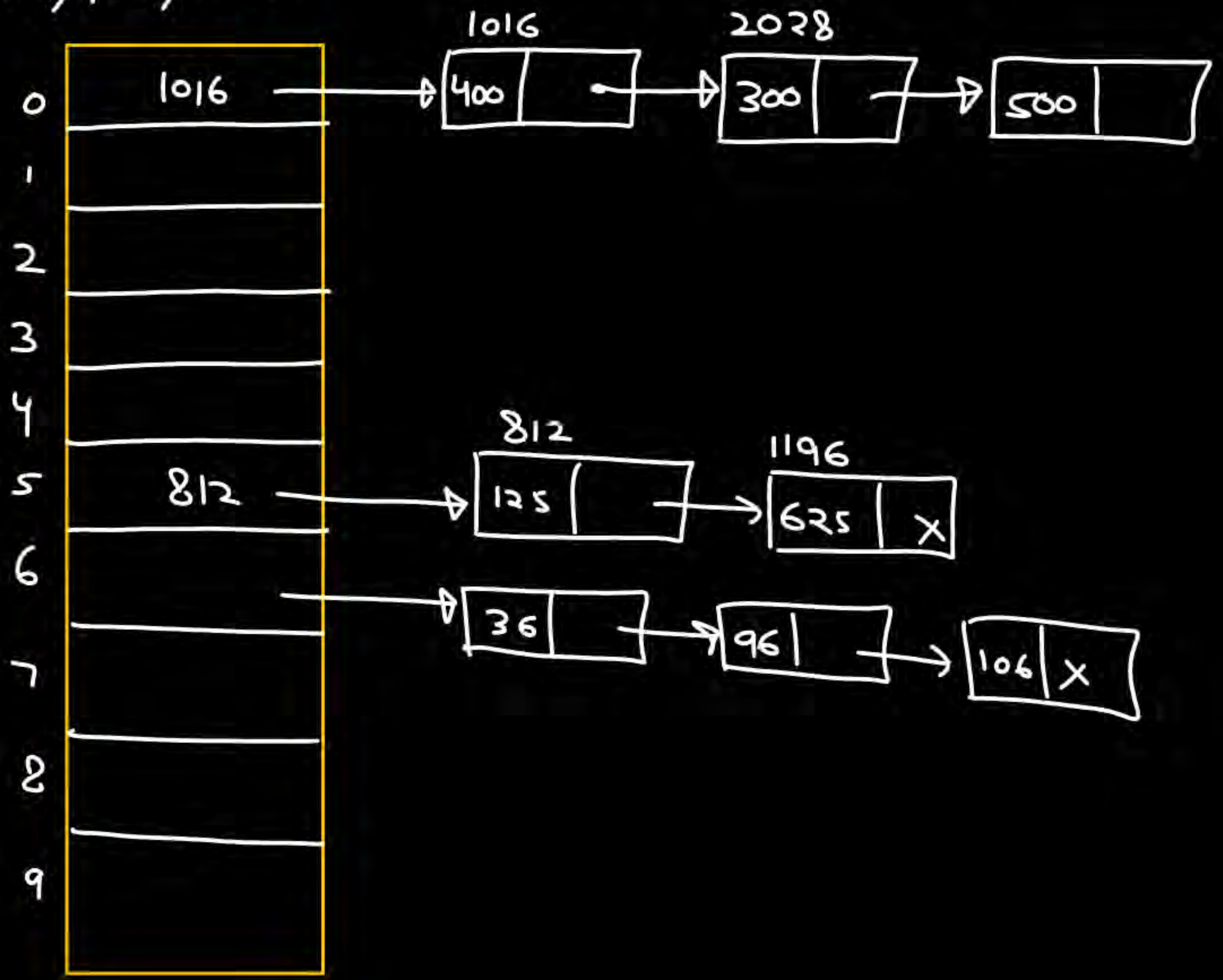


Keys: 400, 300, 125, 625, 36, 96, 106, 500

m=10

worst case → all Keys will map to same bucket

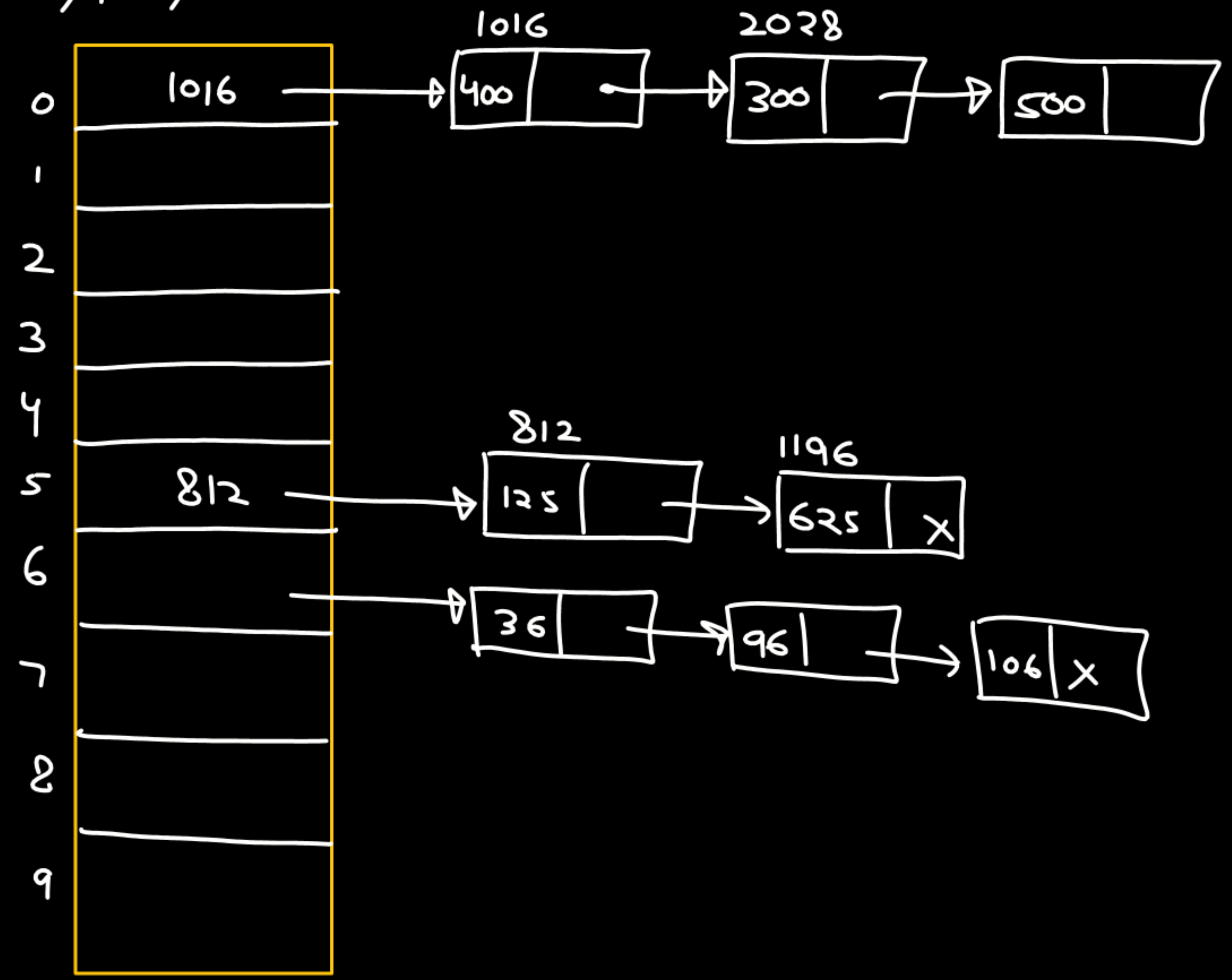
700, 800, 900, 600, 60, 70



Keys: 400, 300, 125, 625, 36, 96, 106, 500

m=10

AVL-tree



# Deletion

$h(K) = K \bmod m$

$m = 12$

Keys: 31, 26, 43, 27, 34, 12, 46, 14, 58

31 → 7, 26 → 2, 43 → ~~7~~ 8, 27 → 3, 34 → 10, 12 → 0, 46 → ~~10~~ 11, 14 → ~~2~~ ~~3~~ 4, 58 → 1

| | |
|---|---|
| 0 | |
| 1 | |
| 2 | |
| 3 | |
| 4 | |
| 5 | |
| 6 | |
| 7 | |
| 8 | |
| 9 | |
| 10 | |
| 11 | |

## Deletion

$h(K) = K \bmod m$

$m = 12$

Keys : 31, 26, 43, 27, 34, 12, 46, 14, 58

7, 2, 8, 3, 10, 0, 11, 4, 1

(i) delete 26

(ii) Search 14

| | |
|---|---|
| 0 | 12 |
| 1 | 58 |
| 2 | ~~26~~ |
| 3 | 27 |
| 4 | 14 |
| 5 | |
| 6 | |
| 7 | 31 |
| 8 | 43 |
| 9 | |
| 10 | 34 |
| 11 | 46 |

# Deletion

$h(K) = K \bmod m$

$m = 12$

Keys: 31, 26, 43, 27, 34, 12, 46, 14, 58

7, 2, 8, 3, 10, 0, 11, 4, 1

(i) delete 26

(ii) Search 14

$h(14) = 14 \bmod 12$
$= 2$

Re-hash all remaining keys

| | |
|---|---|
| 0 | 12 |
| 1 | 58 |
| 2 | Free |
| 3 | 27 |
| 4 | 14 |
| 5 | |
| 6 | |
| 7 | 31 |
| 8 | 43 |
| 9 | |
| 10 | 34 |
| 11 | 46 |

Deletion - Easy in separate

## Gate-2015

Consider a double-hashing, in which the primary hash fun. $h(k) = k \bmod 23$. and sec. hash func. $h_2(k) = 1 + (k \bmod 19)$

$\boxed{m = 23}$.

Then the address returned by probe 1 in the probe seq. (assume probe seq. begins at probe 0) for key $k = 90$ is ________

$i = 1$

$m = 23$

$h_1(k) = k \bmod 23$

$h_2(k) = 1 + (k \bmod 19)$

$$\boxed{H(k, i) = (h_1(k) + i \cdot h_2(k)) \bmod m}$$

$\boxed{i = 1}$

$k = 90$

$h_2(90) = 1 + 90 \bmod 19 = 15$

$h_1(90) = 90 \bmod 23 = 21$

$H(k, 1) = (21 + 1 \cdot 15) \bmod 23$

$= 36 \bmod 23 = 13$

Gate 2015 - 1M

2. Given a hash table T with 25 slots that store 2000 elements, the load factor $\alpha$, is ________

$$\alpha = \frac{2000}{25} = 80$$

Gate - 2015 (2M)

Q. Which one of the following hash func. on integers will distribute keys most uniformly over 10 buckets numbered 0 to 9 for i ranging from 0 to 2020?

a) $h(i) = i^2 \bmod 10$

b) $h(i) = i^3 \bmod 10$

c) $h(i) = (11 \times i^2) \bmod 10$

d) $h(i) = (12 \times i) \bmod 10$

| | a) | ✓ b) | c) | (d) |
|---|---|---|---|---|
| 0 | 0 | 0 | same as option a | odd numbered bucket ↓ Empty |
| 1 | 1 | 1 | | |
| 2 | 4 | 8 | | |
| 3 | 9 | 7 | | |
| 4 | 6 | 4 | | |
| 5 | 5 | 5 | | |
| 6 | 6 | 6 | | |
| 7 | 9 | 3 | | |
| 8 | 4 | 2 | | |
| 9 | 1 | 9 | | |

# Gate-2014 (2 M)

Consider a hash table with 100 slots. Collisions are resolved using $\boxed{\text{chaining}}$. Assuming simple uniform hashing what is the probability that the first 3 slots are unfilled after first 3 insertions.

A) $(97 \times 97 \times 97)/100^3$

B) $(99 \times 98 \times 97)/100^3$

C) $(97 \times 96 \times 95)/100^3$

D) $(97 \times 96 \times 95)/(3! \times 100^3)$

## Gate-2014 (2M)

Consider a hash table with 9 slots. The hash function is $h(k) = k \bmod 9$. Collisions are resolved by chaining.

Keys : 5, 28, 19, 15, 20, 33, 12, 17, 10.

The max., min. and average chain lengths in hash table respectively are

a) 3, 0 and 1

b) 3, 3 and 3

c) 4, 0 and 1

d) 3, 0 and 2

$h(5) = 5 \bmod 9 = 5$

$h(28) = 28 \bmod 9 = 1$

$h(19) = 19 \bmod 9 = 1$

$h(15) = 15 \bmod 9 = 6$

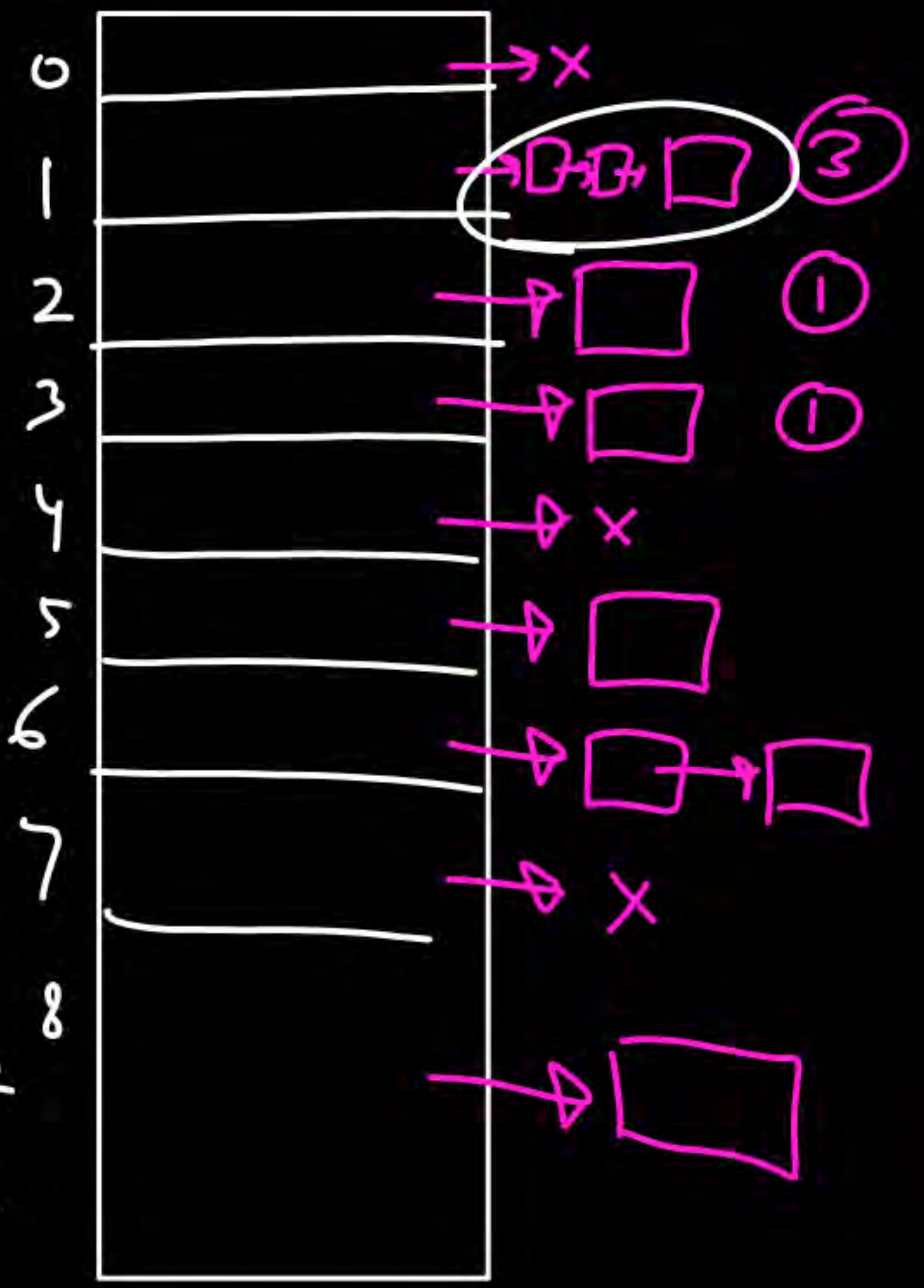
$h(20) = 20 \bmod 9 = 2$

$h(33) = 33 \bmod 9 = 6$

$h(12) = 12 \bmod 9 = 3$

$h(17) = 17 \bmod 9 = 8$

$h(10) = 10 \bmod 9 = 1$




$$avg = \frac{0+3+1+1+0+1+2+0+1}{9} = 1$$

Gate-2010 2M

A hash table of length 10 uses open addressing with $h(K) = K \bmod 10$ and linear probing. After inserting 6 values into an empty hash table, the table is shown below

| | |
|---|---|
| 0 | |
| 1 | |
| 2 | 42 |
| 3 | 23 |
| 4 | 34 |
| 5 | 52 |
| 6 | 46 |
| 7 | 33 |
| 8 | |
| 9 | |

52

42, 52 → same

Which one of the following choices gives a possible order in which the key values could have been inserted in table?

~~a)~~ 46, 42, 34, 52, 23, 33

~~b)~~ 34, 42, 23, 52, 33, 46

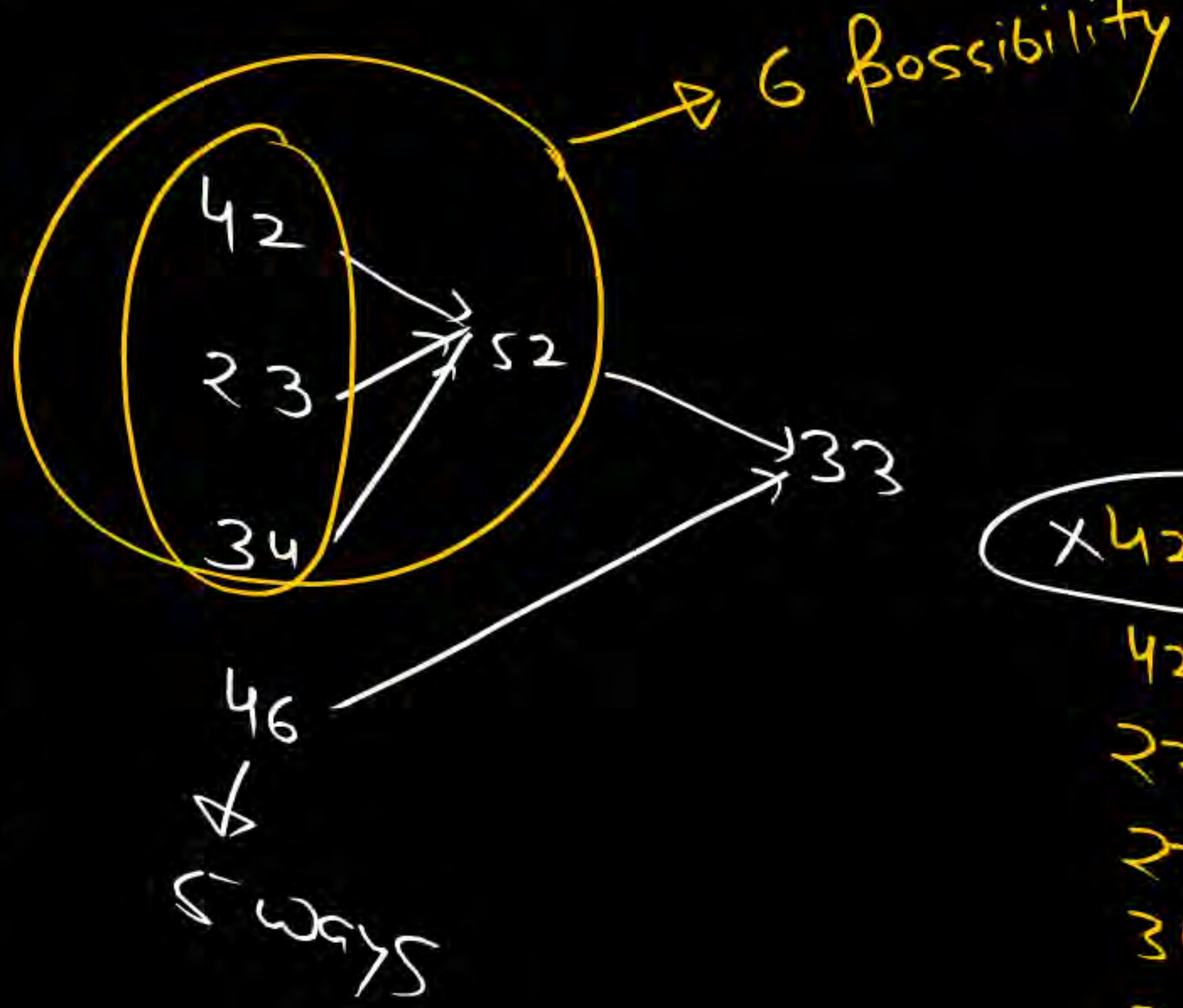
c) 46, 34, 42, 23, 52, 33

~~d)~~ 42, 46, 33, 23, 34, 52

How many diff. insertion sequences of the key values using same h(k) & linear probing will result in the hash table

a) 10
b) 20
c) 30
d) 40

6 × 5
= 30




| | |
|---|---|
| 0 | |
| 1 | |
| 2 | 42 |
| 3 | 23 |
| 4 | 34 |
| 5 | 52 |
| 6 | 46 |
| 7 | 33 |
| 8 | |
| 9 | |

52

42, 52
↳ same

~~42~~, ~~23~~, ~~34~~, ~~52~~
42, 34, 23, 52
23, 34, 42, 52
23, 42, 34, 52
34, 23, 42, 52
34, 42, 23, 52

Consider a hash table of size 11 that uses open addressing with linear probing. Let $h(k) = k \bmod 11$ be the hash function used.

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

A seq. of records with keys 43, 36, 92, 87, 11, 4, 71, 13, 14 is inserted into initially Empty table, the bins of which are indexed from 0 to 10. What is the index of the bin into which the last record is inserted?

a) 2

b) 4

c) 6

d) 7

Happy Learning

1st insert $\Rightarrow$

$$\frac{97}{100}$$

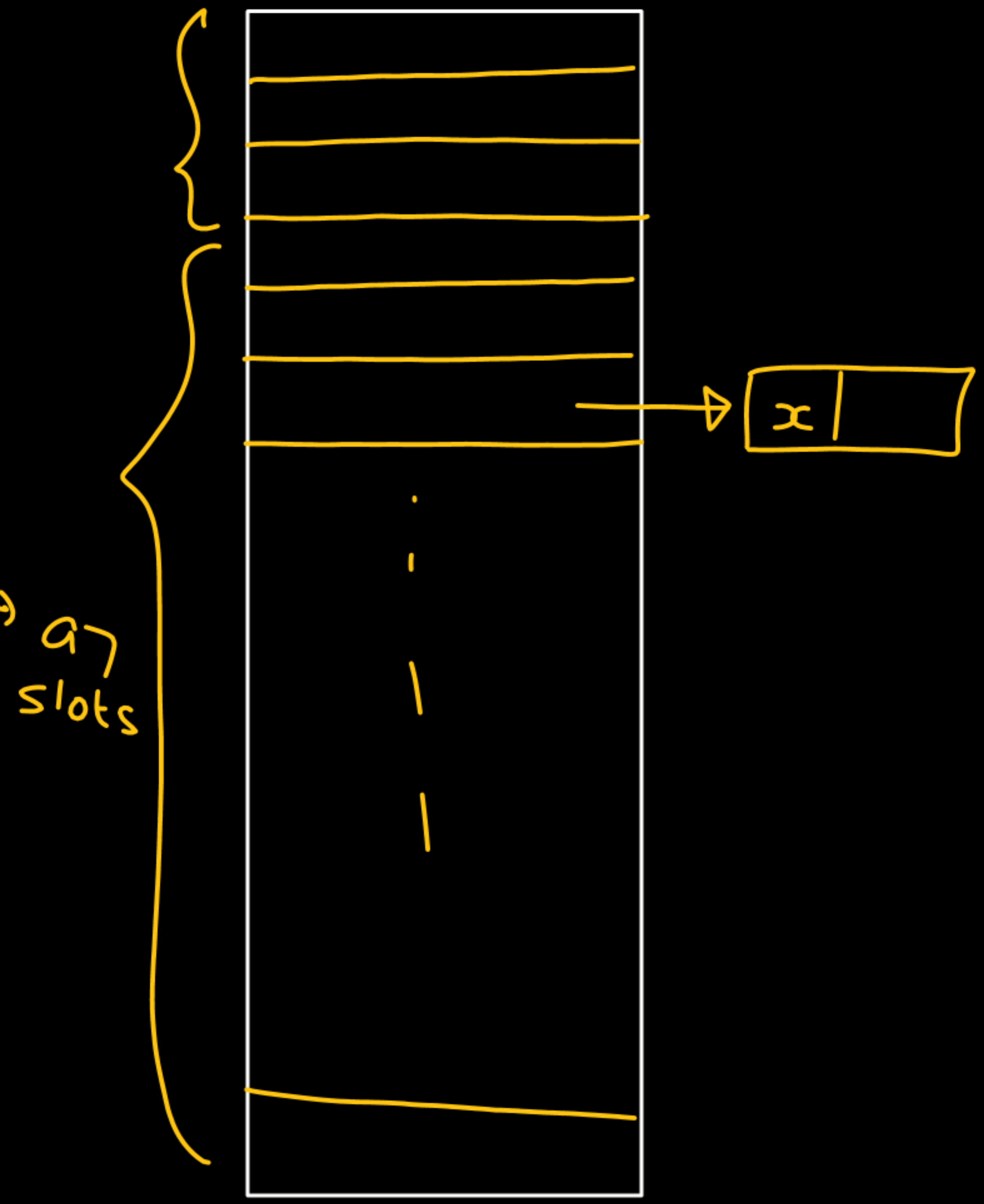
2nd insert

$\hookrightarrow \frac{97}{100}$

3rd insert

$\hookrightarrow \frac{97}{100}$

$$\frac{97}{100} \times \frac{97}{100} \times \frac{97}{100}$$




THANK YOU GW
SOLDIERS !