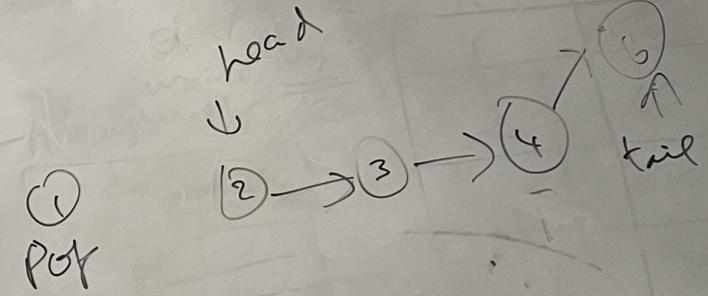


SMART

kapp

- XIFO
- Queue using stacks
- | | |
|-----------|--------|
| - Push | List |
| | $O(1)$ |
| - Pop | $O(n)$ |
| - Peek | $O(1)$ |
| - isEmpty | $O(1)$ |

$[x, 2, 3, 4]$



①
POF

$l = []$

$q = deque()$

Stack & Queue
- Dynamic DS

Amortized TC

Populated

1

2

Decrease

1

2

3

4

POP

5

POP

POP

POP

7

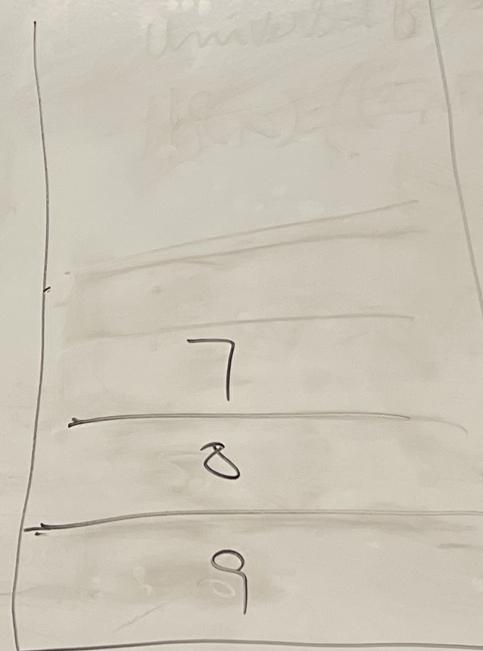
8

9

Peek

IN

Push



OUT

POP

O(1)

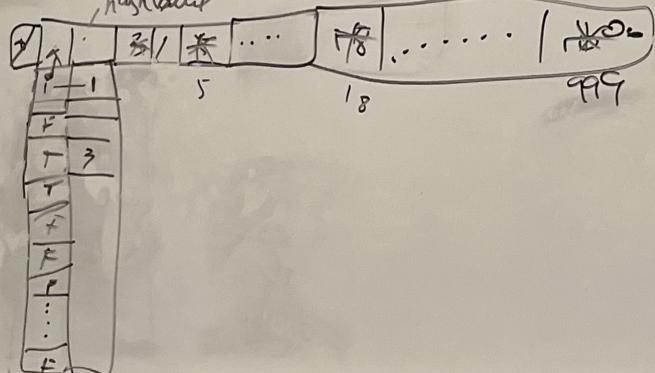
- HW
1. Does size of list shrink in size? Yes
 2. List and Stack? Are they same? Yes
- HW

[1, "Swathy", [→], 2]

- Key in array is index $l = [1, 2, 1, 3, 5, 2, 4]$
 $\text{Print}(\text{set}(l))$

Hashset

[1, 2, 3, 18, 1000, 5] $O(1)$ $\frac{300}{1000}$



$\underline{300} / \underline{1000}$

$\underline{100} / \underline{1000}$

$\underline{300} / \underline{1000}$

2^{32} Hashfn.

$300 / \underline{1000}$

List (Dynamic Array)

→ How? HW.

- Universal function

- $Hf(x) = ((ax_1) + (bx_2)) \times \frac{1}{\text{size of array}}$

- $Hf_2(x) = x \% \text{ size of array}$

if s.bucket(h):

- HashSet
- Overview of Hashmap
- Queue using Stack (2)

- Internally Hashmap is array (2)
- Hashmap and Hashtable? \rightarrow Hw
- HashSet?
- Set in python?
- Collision?

2 Keys, same hash value.

Collision Handling Technique?

- Linear Chaining
 - Double Hashing
 - Linear Probing
- $\rightarrow \Theta(1)$ access/Search

$$\begin{array}{|c|c|} \hline 0 & 1000000 \\ \hline 100 & 10000 \\ \hline \end{array}$$

$$\begin{array}{|c|c|} \hline 0 & 32 \\ \hline 25 \\ \hline 1000 \times 1000 \\ \hline 10^6 + 1 \\ \hline \end{array}$$

why?
 $\Theta(1)$ search
 using key

- Arrays are Motivation



[1, 2, 1000, 3, 18]

- Sing Major Factor $\Theta(1)$

- Equally distributing.

Hashmap

K	V
1	90
2	80
3	75
4	80
5	72
6	80

8
7
6
5

List (Dynamic Array)

1. HW Does size of list shrink in size? Yes
2. list and Stack? Are they same? Yes

HW

$[1, "Swathy", [\rightarrow], 2]$
 - Key in array is index $l = [1, 2, 1, 3, 5, 2, 4]$
 $\text{Print}(\text{set}(l))$

HW?

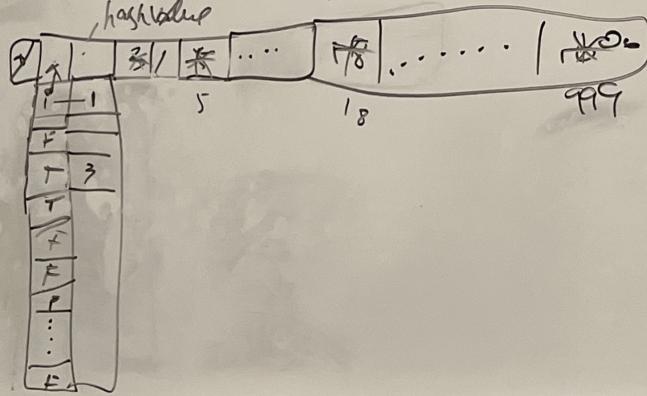
- Universal function

$$- Hf(x) = ((ax + b) \times R_1 + c) \times R_2$$

$$- Hf_2(k) = k \% \text{ size of array}$$

Hashset

$[1, 2, 3, 18, 1000, s]$ $O(1)$ $\frac{300}{1000}$



$$\frac{300}{1000} \% 1000$$

$\frac{300}{1000} \% 1000$

2^{nd} Hashfn.

$$\frac{300}{1000} \% 1000$$