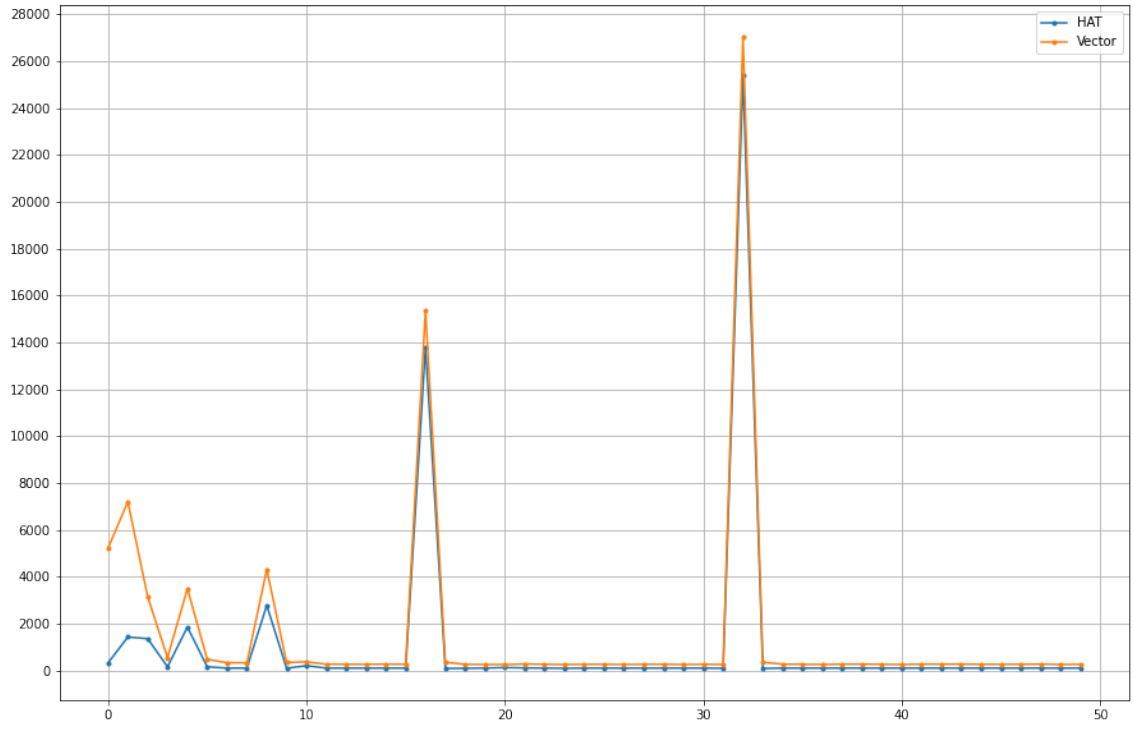
ICCS315: Assignment 1

Vanessa Rujipatanakul

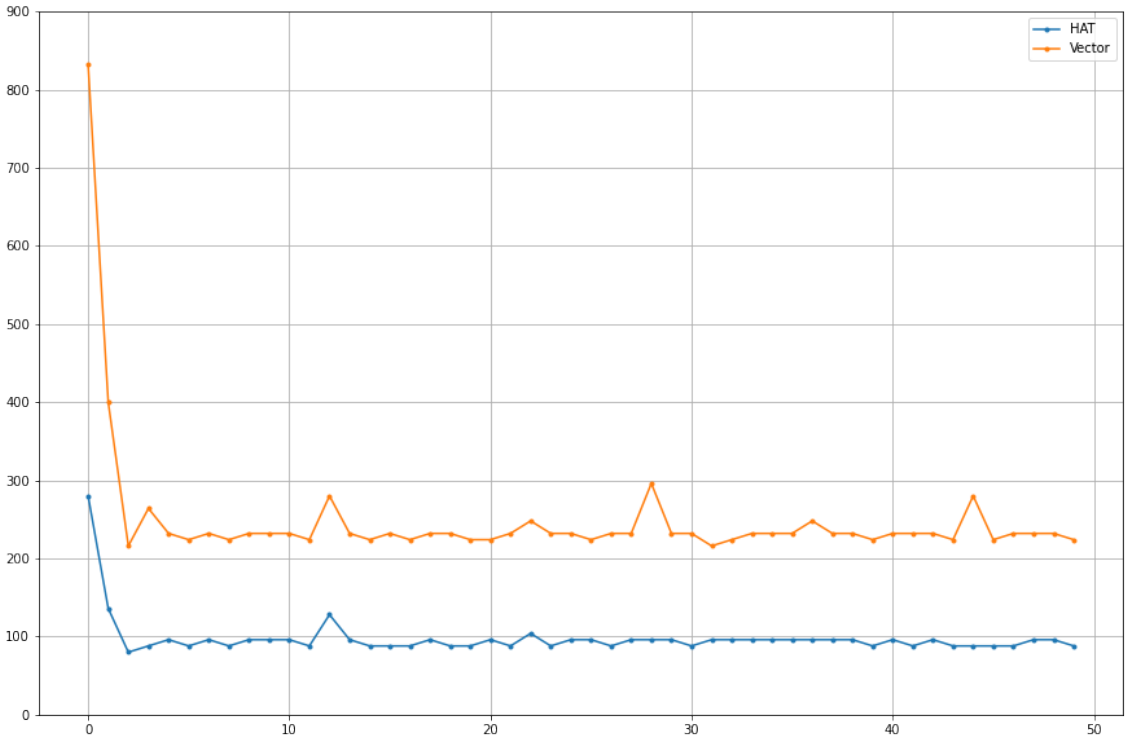
30/01/2023

1: *Resizable Arrays*

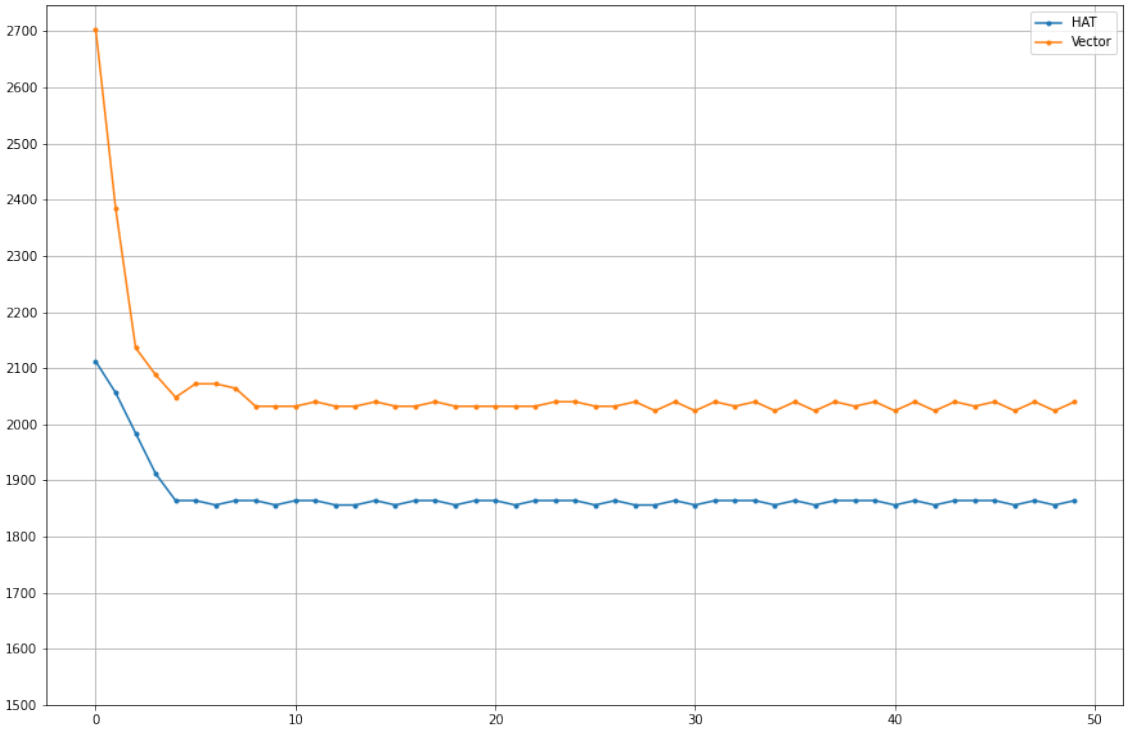
(a) Append Latency
HAT $\approx$ 96 cycles
Vector $\approx$ 260 cycles



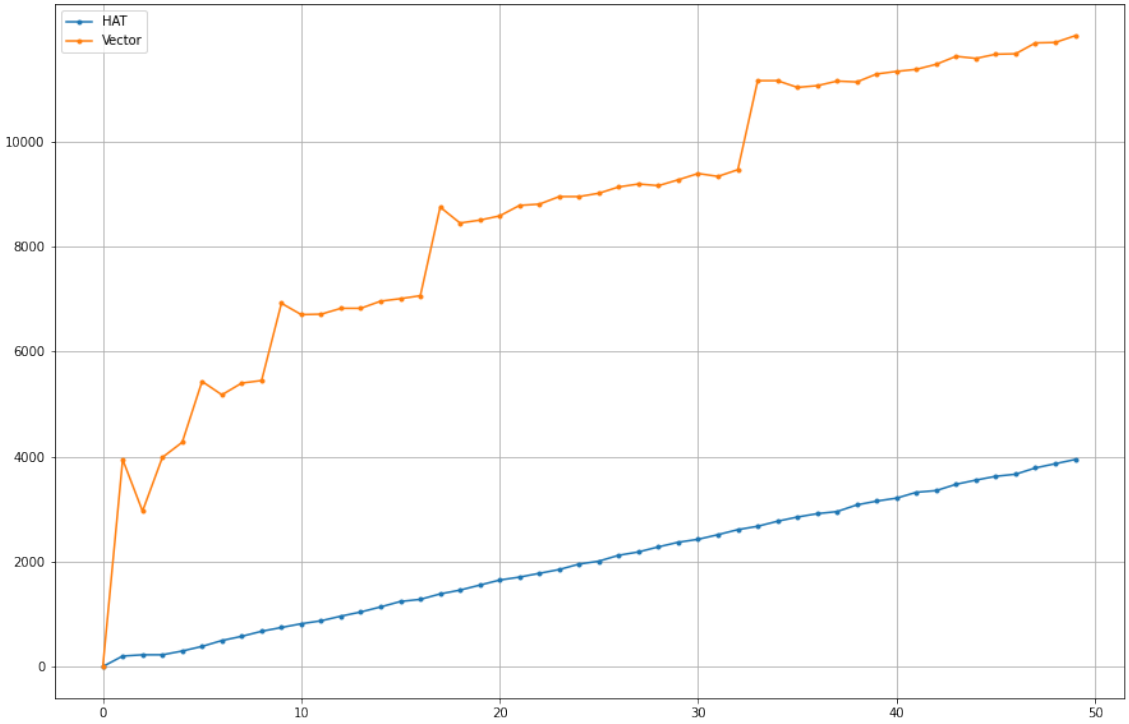
(b) Access Latency
HAT $\approx$ 96 cycles
Vector $\approx$ 232 cycles



(c) Scan Latency
HAT ≈ 1864 cycles
Vector ≈ 2036 cycles

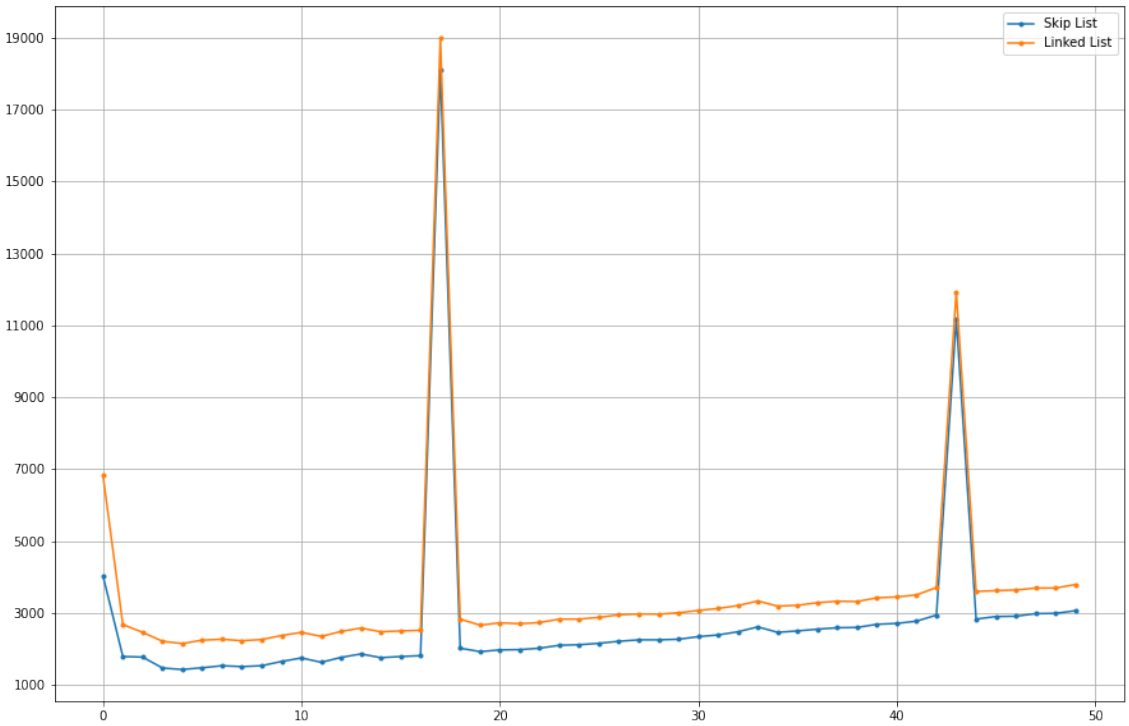


(d) Overall Latency
HAT ≈ 1980 cycles
Vector ≈ 8984 cycles

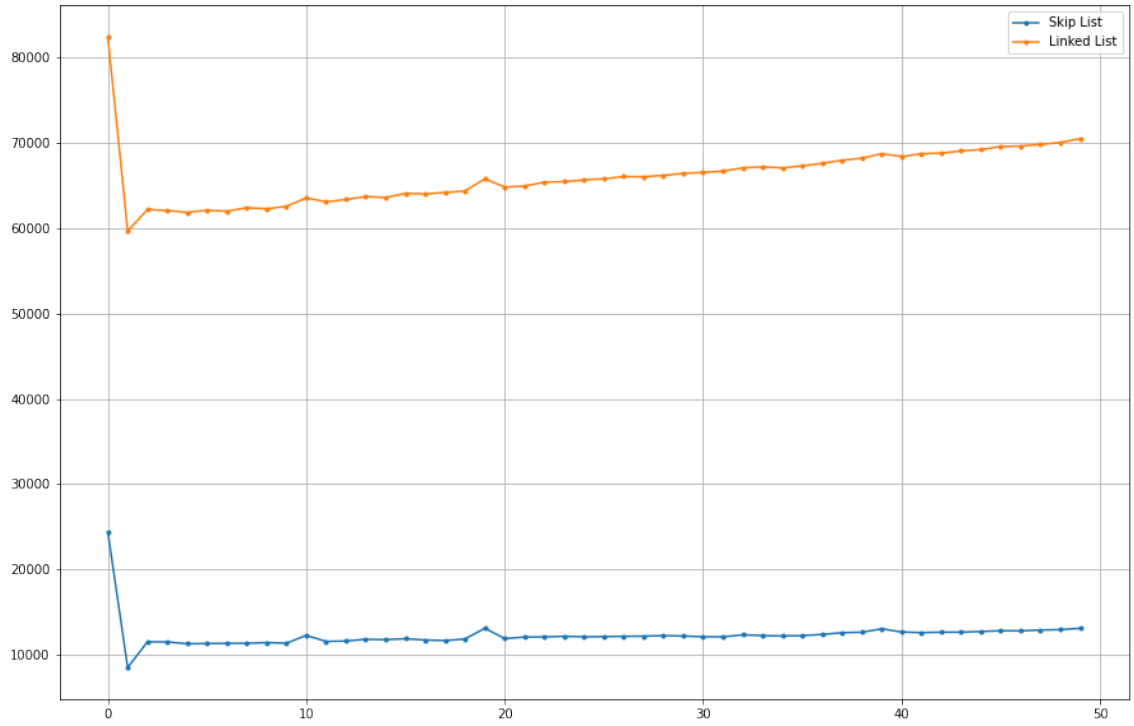


3: *Skip Lists*

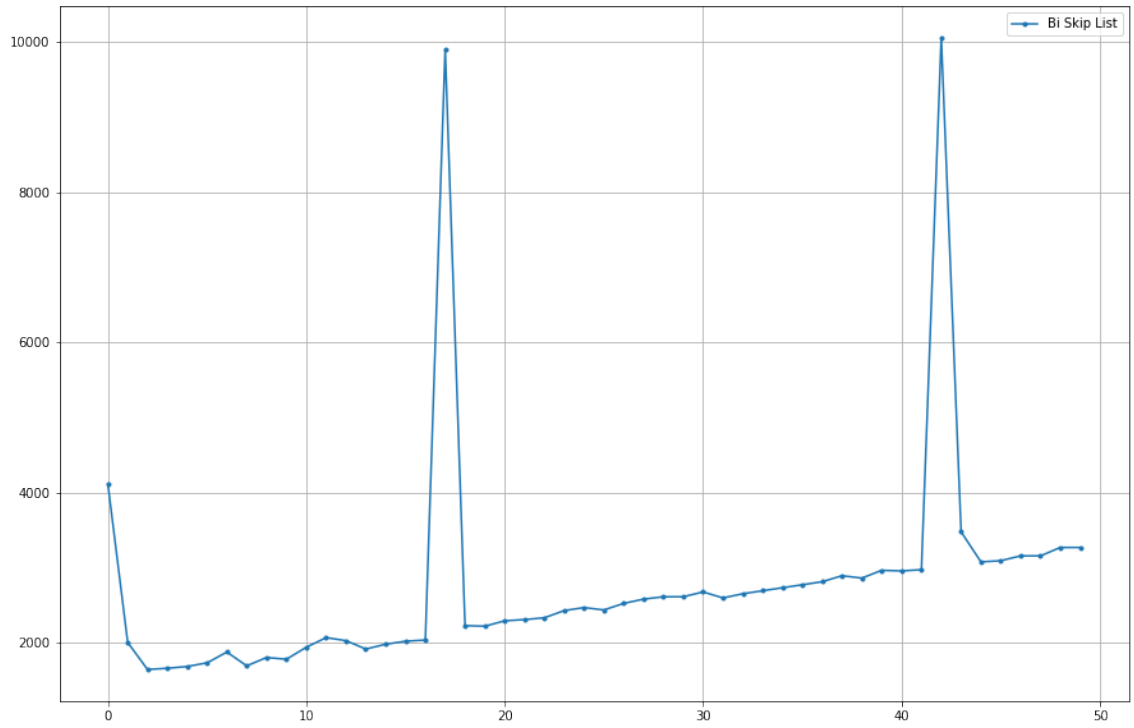
(a)
Insert Latency
Skip List ≈ 2236 cycles
Linked List ≈ 2960 cycles



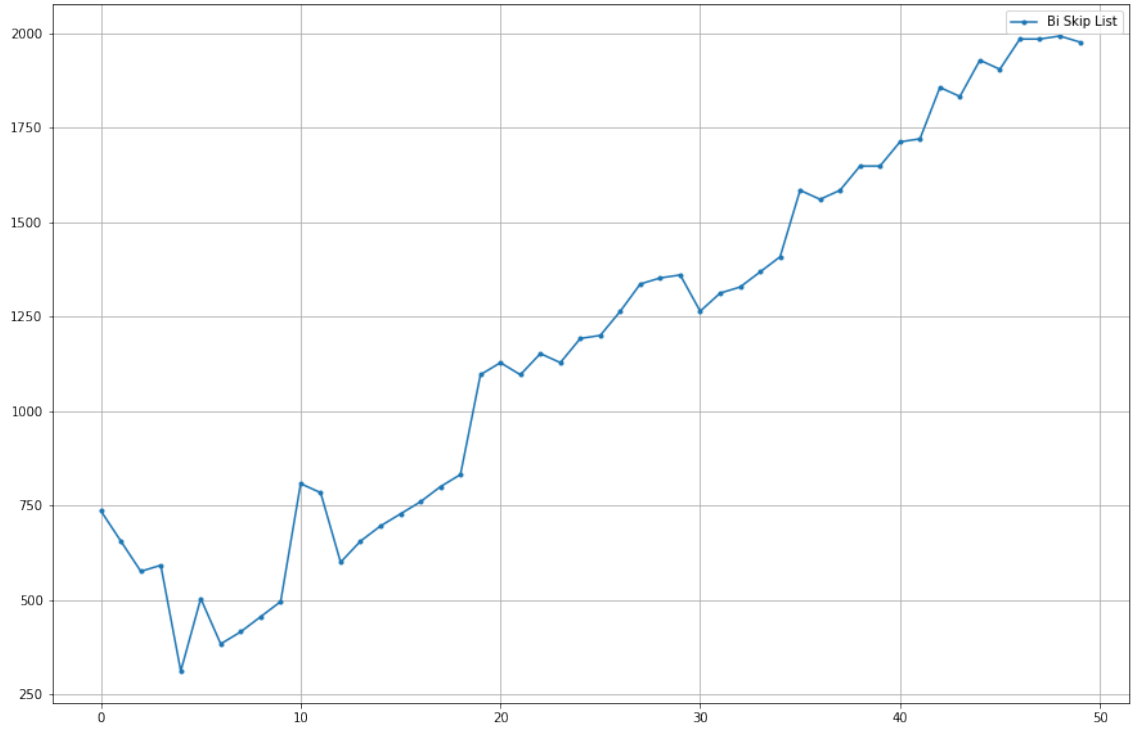
Search Latency
Skip List ≈ 12212 cycles
Linked List ≈ 65856 cycles



(b) Insert Latency ≈ 2556 cycles

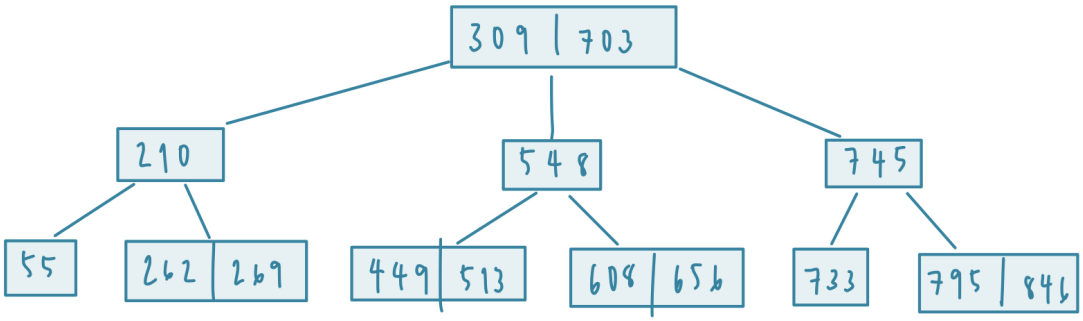


Search Latency ≈ 1196 cycles

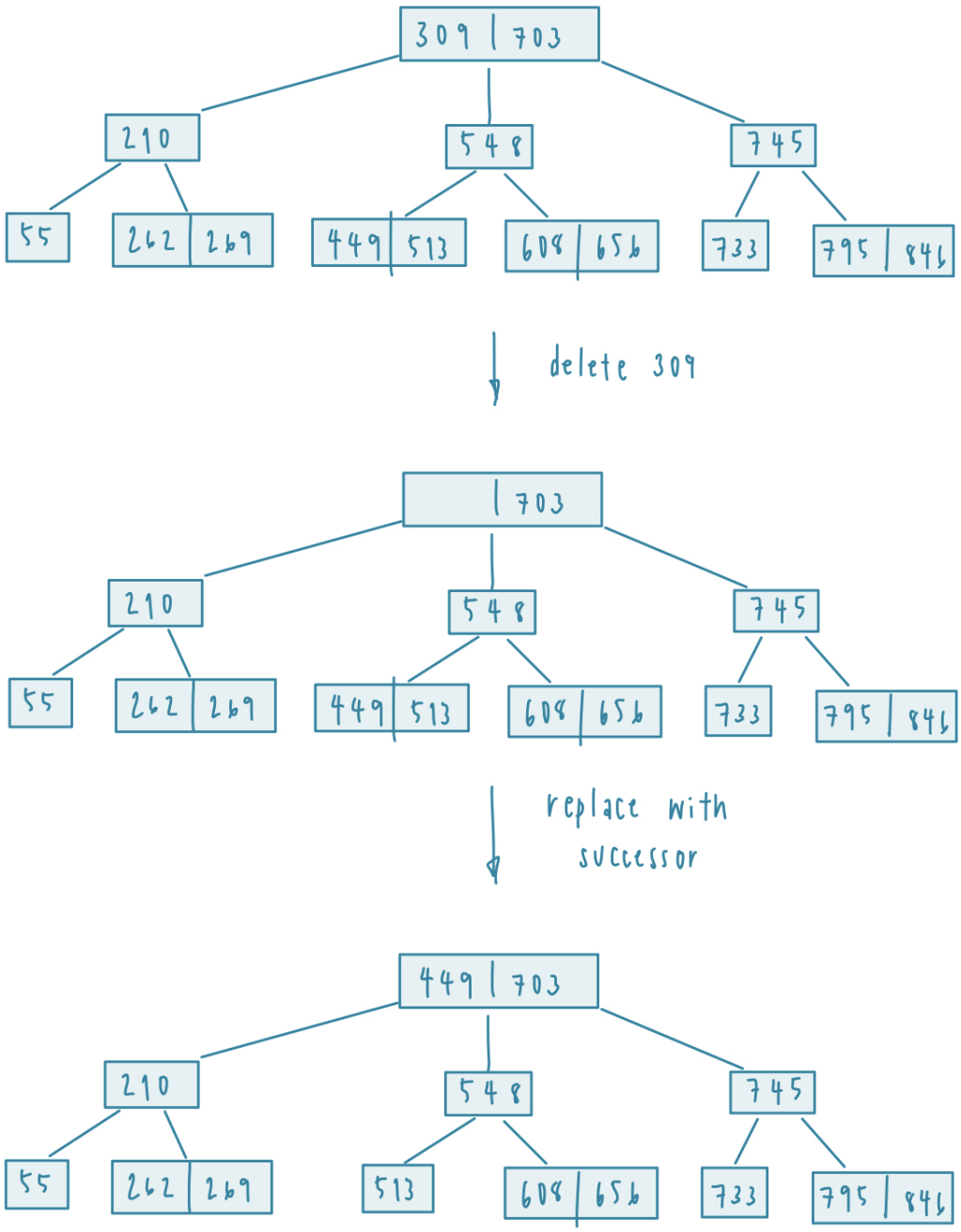

Bi Skip List
2000
1750
1500
1250
1000
750
500
250
0
10
20
30
40
50

4: *(a, b) tree*

(a)


309 | 703
210
548
745
55
262 | 269
449 | 513
608 | 656
733
795 | 846

(b)



4: ***B-Tree Speed***

An optimal value of b should be between 100 and 1000. This range seems to be the balance between minimizing the number of disk accesses and minimizing the space overhead per node. A smaller value of b means that each node contains fewer keys and values, reducing the space overhead per node. But this also increases the height of the tree, leading to more disk accesses and slower performance for operations that require disk access. A larger value of b means that each node contains more keys and values, reducing the height of the tree and the number of disk accesses required. However, this also increases the space overhead per node.