Skip list

Okay! Let's explain this in a very simple way.

This is about a Skiplist – a special data structure that works like a linked list + shortcuts for faster search.

✅ What is a Skiplist?

- Imagine a normal sorted linked list → slow search because you go one by one.
- Skiplist adds extra "fast lanes" (levels) so you can skip multiple nodes at once.
- Each node can appear on 1 or more levels randomly.

So operations like search, add, erase become O(log n) instead of O(n).

✅ Step-by-step what's happening

We start with an empty Skiplist.

1️⃣ `skiplist.add(1)`

- Add 1 to the list
- It's inserted at some random levels (maybe level 1, maybe higher)

Now skiplist has:

Copy Edit

```
1
```

2️⃣ `skiplist.add(2)`

- Add 2
- Also placed at random levels (but always after 1, because it's sorted)

Now:

Copy Edit

```
1   2
```

3️⃣ skiplist.add(3)

- Add 3
- Goes after 2

Now:

```
1   2   3
```

4️⃣ skiplist.search(0) → False

- Search for 0
- Traverse skiplist, but 0 < 1, so not found → returns False

5️⃣ skiplist.add(4)

- Add 4
- Goes after 3

Now:

```
1   2   3   4
```

6️⃣ skiplist.search(1) → True

- Search for 1
- Found in skiplist → returns True

7️⃣ skiplist.erase(2) → True

- Removes 2 from all levels

- After erase:

```
1  3  4
```

- Returns True because it existed and got removed.

8️⃣ `skiplist.search(2)` → `False`

- Search for 2
- Not found anymore (we erased it) → returns False

✅ So the final Skiplist

After all operations:

```
1  3  4
```

- `0` never existed
- `2` was erased
- `1, 3, 4` remain

✅ Why random levels?

Each node randomly decides how many levels it appears on.

- Higher levels = fewer nodes = fast jumps
- Lower levels = all nodes = exact order

So you get logarithmic search speed similar to a balanced BST but easier to implement.

Would you like me to:

✅ Draw a visual skiplist with multiple levels showing how 2 gets erased?
✅ Or give a real-world analogy (like express lanes on a highway)?

Got it! Let's explain how adding works in a Skiplist with a simple example.

✅ Current Skiplist

Right now it contains sorted numbers:

```
30  40  50  60  70  90
```

Since it's a Skiplist, it's not just a single linked list — it has multiple levels:

```pgsql
Level 3:  30 ---------- 60 ---------- 90
Level 2:  30 -------- 50 -------- 70 -------- 90
Level 1: 30  40  50  60  70  90
```

(Higher levels have fewer "express" nodes; Level 1 has all nodes.)

✅ We want to add 80 and 45

Let's insert them one by one.

1️⃣ Adding 80

- Skiplist is sorted, so 80 should go between 70 and 90.

Steps:

1. Start from the top level, move forward until just before 90.
2. Drop down a level and do the same.
3. Finally, at level 1, insert `80` before `90`.

So now the bottom level is:

```
30  40  50  60  70  80  90
```

- Random levels will decide if `80` also appears in level 2 or 3 (like a coin toss).
  Maybe:

```pgsql
Level 3:  30 ---------- 60 ---------- 90
Level 2:  30 -------- 50 -------- 70 -------- 80 -------- 90
```

```
Level 1: 30   40   50   60   70   80   90
```

2️⃣ Adding 45

- Should go between 40 and 50.

Steps:

1. Start at the top level, move past `30` and `40`, stop before `50`.
2. Drop down, do the same.
3. Insert `45` before `50` at level 1.

Now the bottom level is:

```
30   40   45   50   60   70   80   90
```

Again, random levels decide if it appears in higher levels. Maybe:

```pgsql
Level 3:     30 ------------------ 60 ------------------ 90
Level 2:     30 -------- 50 -------- 70 -------- 80 -------- 90
Level 1: 30   40   45   50   60   70   80   90
```

Or it could also appear on level 2 if the randomization allows:

```pgsql
Level 2:     30 ---- 45 ---- 50 ---- 70 ---- 80 ---- 90
```

✅ Final Skiplist after insertion

Now it has:

```
30   40   45   50   60   70   80   90
```

And its higher levels still keep shortcuts for faster search.

✅ Key idea

- Search for where the new value fits (like binary search in a layered linked list).

- Insert it in the correct position at level 1.
- Randomly promote it to higher levels.

That's why Skiplist stays balanced on average.

Would you like me to draw a visual diagram with multiple levels showing BEFORE and AFTER inserting 80 & 45? Or also explain how the random coin toss promotion works?