Explain about Cache Replacement Algorithm (FIFO, LRU, LFU)

1. FIFO (First-In-First-Out)

- **How it works:** The cache keeps a queue of items in the order they were added. When new data comes in and the cache is full, the *oldest* item (the one that entered first) is removed to make space for the new one.
- **Example:** Imagine a cache that can hold 3 items. If you add A, B, C, and then need to add D (cache is full), A (the oldest) is removed, so cache becomes B, C, D.
- **Pros:** Very simple and easy to implement.
- **Cons:** Doesn't consider if an old item is still being used often. Frequently used items might get evicted just because they are old.[1] [2] [3] [4]

2. LRU (Least Recently Used)

- **How it works:** When the cache is full and you need to add something new, the cache throws out the item that hasn't been used for the longest time. Every time an item is used, it's marked as most recently used.
- **Example:** If items A, B, and C are in the cache and you access B, B becomes most recently used. If you then add D and need to evict one, A or C (the least recently used) will go, depending on which was used less recently.
- **Pros:** Good for patterns where recently used items are likely to be used again.
- **Cons:** Slightly more complex to implement than FIFO because it must track usage history.[5] [6] [7] [8]

3. LFU (Least Frequently Used)

- **How it works:** The cache always tracks how often each item is used. When space is needed, the item with the *lowest access count* (least frequently used) is removed.
- **Example:** If item A was used 10 times, B 2 times, and C 1 time, adding D would evict C (lowest usage).
- **Ties:** If multiple items have the lowest usage, the oldest is usually kicked out first.
- **Pros:** Good when certain data is accessed much more often than the rest and should be kept longer.

- **Cons:** Harder to implement efficiently. Can sometimes keep old, once-popular data even if it's no longer needed if not updated properly.[2] [9] [10]

Quick Comparison Table

Algorithm	What gets evicted?	Pros	Cons
FIFO	Oldest item in cache	Simple, low overhead	Ignores recent/frequent use
LRU	Least recently used item	Tracks access patterns	Requires tracking usage order
LFU	Least frequently used item	Remembers popularity	Complex, may lag on changes

⁂

1. https://en.wikipedia.org/wiki/Cache_replacement_policies
2. https://algocademy.com/blog/cache-algorithms-fifo-vs-lru-vs-lfu-a-comprehensive-guide/
3. https://www.studocu.com/row/messages/question/3714197/discuss-least-recently-used-lru-first-in-first-out-fifo-and-least-frequently-used-lfu
4. https://www.geeksforgeeks.org/dsa/program-page-replacement-algorithms-set-2-fifo/
5. https://www.geeksforgeeks.org/system-design/lru-cache-implementation/
6. https://www.topcoder.com/thrive/articles/lru-cache
7. https://www.scaler.com/topics/lru-page-replacement-algorithm/
8. https://redis.io/glossary/lru-cache/
9. https://en.wikipedia.org/wiki/Least_frequently_used
10. https://www.geeksforgeeks.org/dsa/least-frequently-used-lfu-cache-implementation/