Algorithm: LFU Page Replacement Simulation

Step 1: Start the program.

Step 2: Initialize two empty lists:

- cache to store pages.
- frequency to store access count of each page.

Step 3: Read the cache size from the user.

Step 4: Read the page reference string (access sequence) from the user.

Step 5: Initialize:

- page_hits = 0
- page_faults = 0

Step 6: For each page in the reference string, do the following:

Step 6.1: If the page is already present in the cache
→ Increment its frequency counter.
→ Increment page_hits.
→ Mark it as a **HIT**.

Step 6.2: Else (page is not in cache)
→ Increment page_faults.
→ Mark it as a **MISS / PAGE FAULT**.

If cache is not full:
→ Insert the page into the cache.
→ Set its frequency to 1.

Else (cache is full):
→ Find the page with the **lowest frequency**.
→ Remove that page from the cache.
→ Insert the new page with frequency 1.

Step 7: Display the cache contents and frequency list after each access.

Step 8: After all pages are processed:

- Calculate hit_ratio = page_hits / total number of references.

Step 9: Display:

- Total Page Hits
- Total Page Misses
- Hit Ratio

Step 10: Stop the program.