

Operating Systems Lab 8

210010020 & 210010006

I. Implementations:

We have implemented following page replacement algorithms for this assignment:

1. **FIFO (First-In, First-Out):**

- Pages are replaced in the order they were brought into memory.
- When a page fault occurs and the memory is full, the oldest page (the one brought in first) is replaced.
- Simple to implement but may not always result in optimal performance.

2. **LRU (Least Recently Used):**

- Pages are replaced based on which one hasn't been used for the longest time.
- When a page fault occurs and the memory is full, the page that was least recently accessed is replaced.
- Requires tracking the access history of each page, typically implemented using a data structure like a queue or a linked list.
- Generally provides better performance than FIFO because it tends to retain pages that are more likely to be reused soon.

3. **Random:**

- Pages are replaced randomly when a page fault occurs and the memory is full.
- Instead of following a specific rule like FIFO or LRU, a page is selected for replacement randomly.
- Simple to implement and doesn't require tracking access history, but may not always perform optimally.
- Offers unpredictability in page replacement, which can sometimes lead to better performance in certain scenarios.

II. To study the behaviour of number of frames with Page Faults

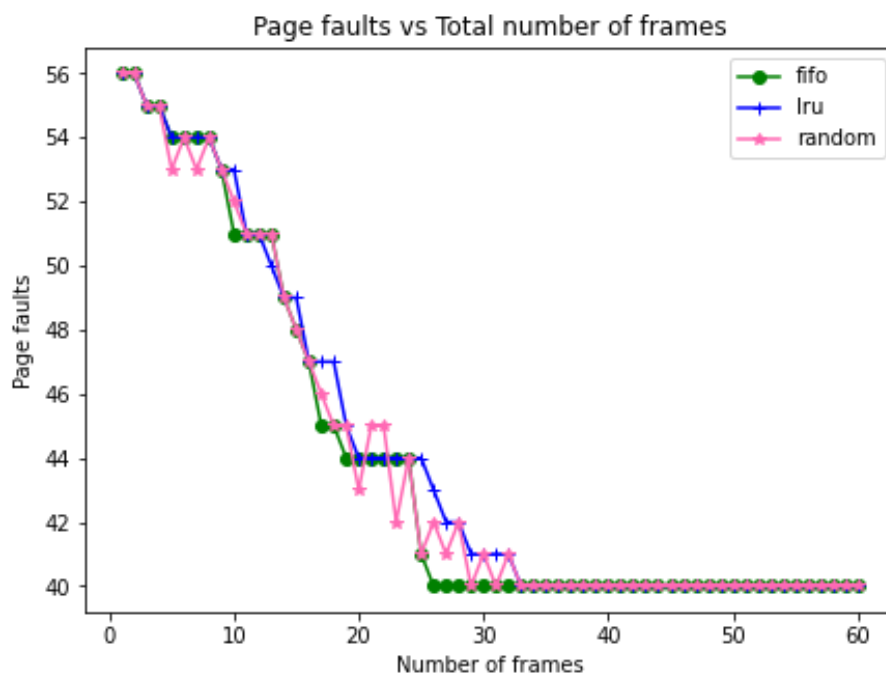
We create 5 workloads of randomly generated values and calculate the total number of misses for each replacement algorithm, then we make a plot for total number of misses with the number of addressable pages.

Following are the observation:

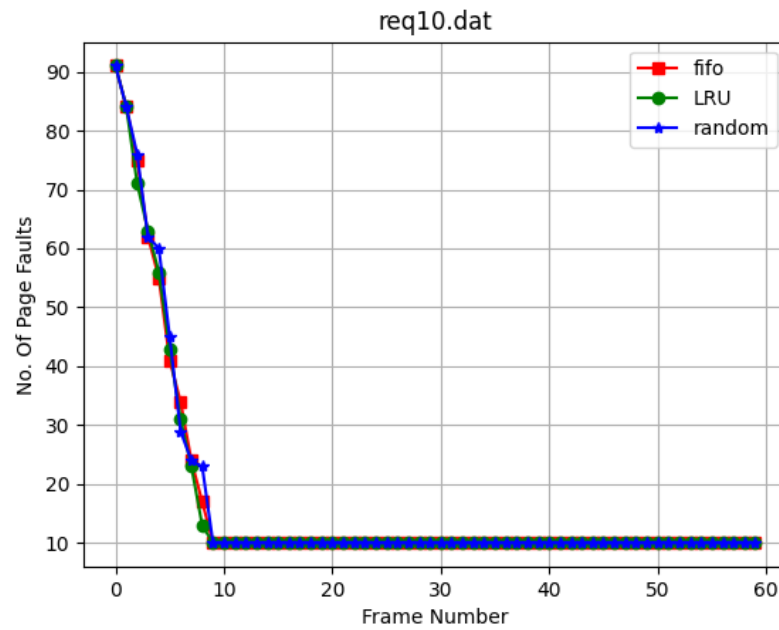
- i. With increase in the frames the Page Fault count usually decreases for all the page replacement algorithm. For number of frames = 1 the Page fault is nearly 100% almost all the algorithms give same results.
- ii. The page fault count remains constant after some point even when frame numbers increased. This shows us that increasing the number of frames can reduce the possibility of Page Faults.
- iii. It is observed that LRU and FIFO show almost same behaviour, with Random policy performing comparatively better than rest in some cases.

III. Plots:

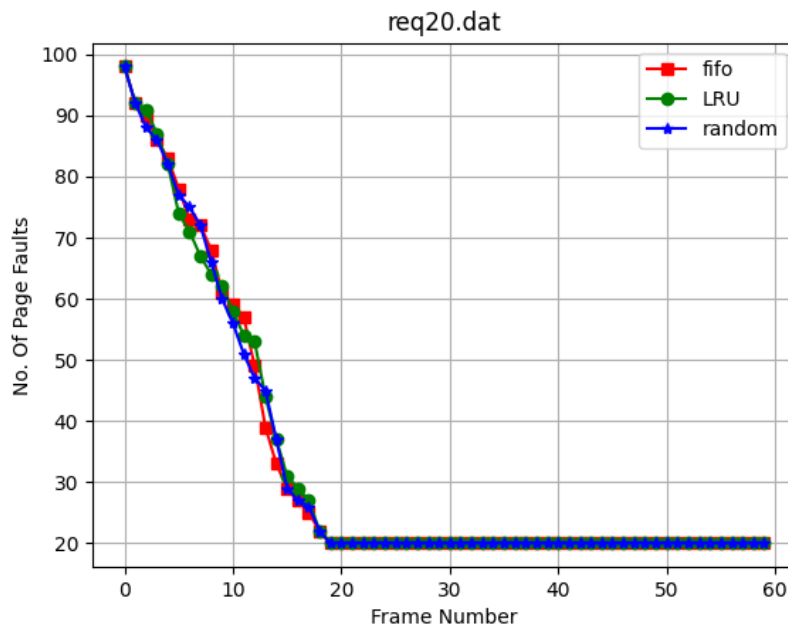
- i. Given req1.dat file total Pages = 56



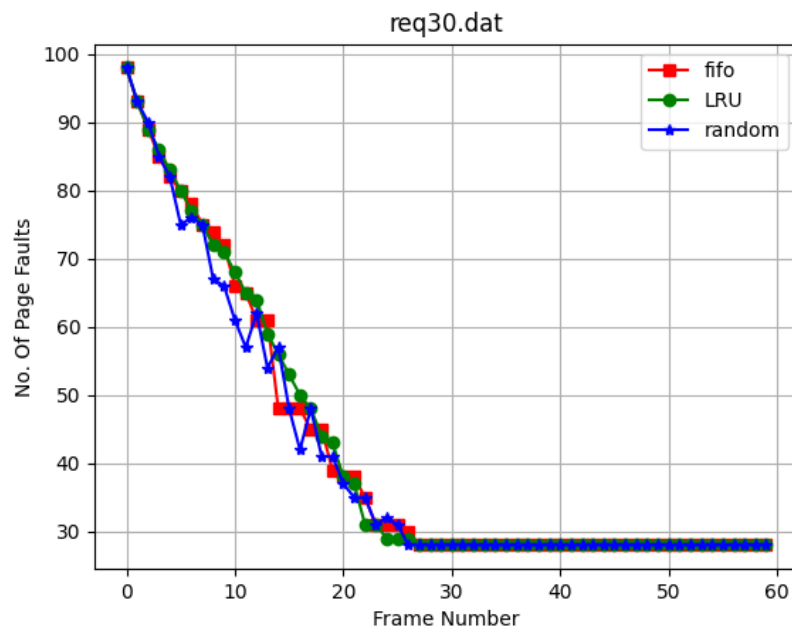
- ii. For req10.dat Totaal 100 pages. Maximum unique page numbers = 10



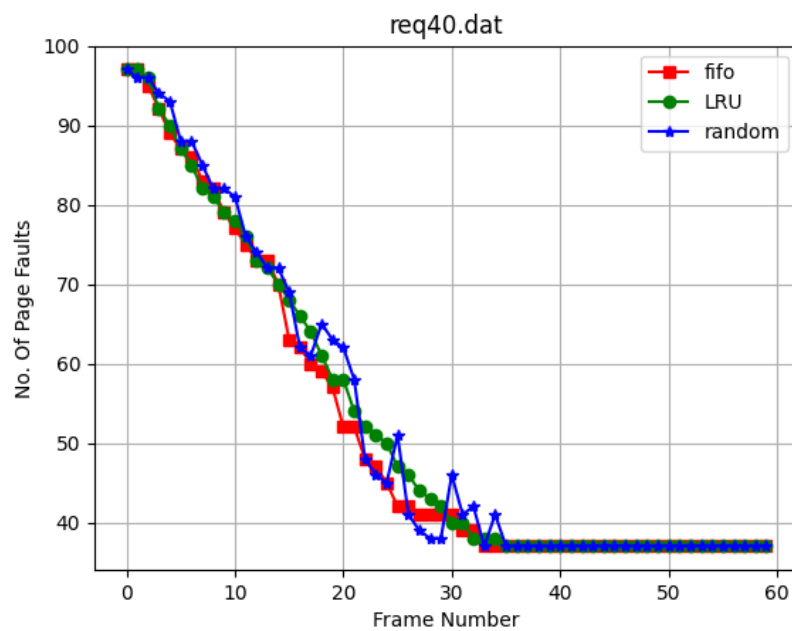
- iii. For req20.dat Total 100 pages. Maximum unique page numbers = 20



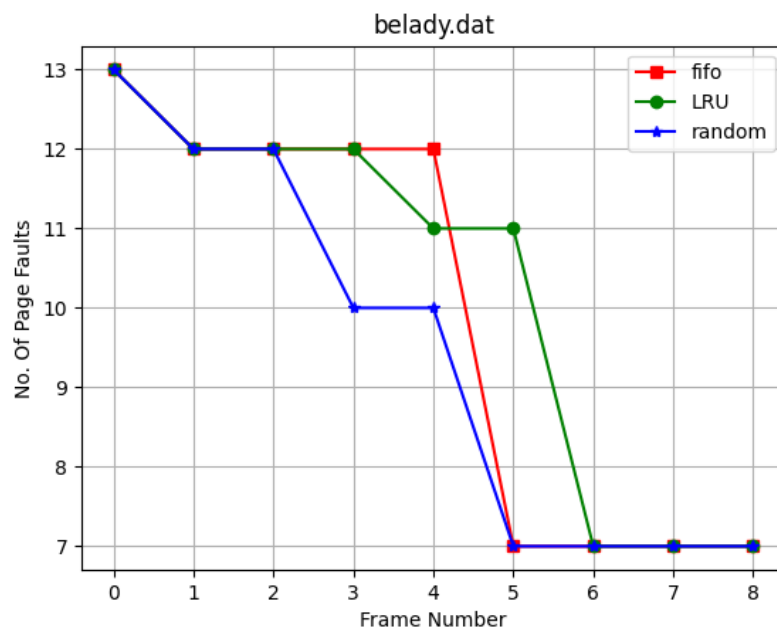
- iv. For req20.dat Total 100 pages . Maximum unique page number = 30



- v. For req20.dat Total 100 pages . Maximum unique page number = 40



- vi. Belady Workload to show that in some cases even though with increase in the number of frames the Page Faults don't reduce.



- vii. 80-20 Workload: In this workload 80% of pages are accessed 20% of times and 20% of pages are accessed 80% of time. In this workload LRU performs better than other policy.

