CS314 : Operating Systems Lab

Lab 8

Sourabh Bhosale
200010004

March 19, 2023

1 Page Replacement policies

FIFO (First In First Out)

Advantages:

- Simple implementation: FIFO is a simple algorithm to implement and requires minimal resources.
- No overhead: FIFO doesn't require any additional overhead such as tracking the frequency of page usage or predicting the future page requests.
- Fairness: FIFO provides fairness in terms of memory allocation to processes as the oldest page gets replaced first.

Disadvantages:

- Poor performance: It suffers from the "Belady's anomaly," which means that adding more memory may lead to more page faults, and the system's performance may degrade.
- Inefficient use of memory: Frequently used pages may get replaced, resulting in more page faults.
- No consideration for page importance: Some pages may be more critical than others, but FIFO treats all pages equally.
- Higher I/O operation overhead: FIFO may lead to higher I/O operation overhead as frequently used pages may need to be swapped in and out of memory, resulting in more disk operations.

LRU (Least Recently Used)

Advantages:

- High hit rate: LRU has a higher hit rate than other page replacement policies, such as FIFO or random, because it takes into account the history of page usage.

- Efficient use of memory: LRU is efficient in terms of memory usage, as it replaces pages that are least recently used, freeing up memory for other pages that are more likely to be used in the future.
- Reduced I/O operations Performance improvement: LRU reduces the number of I/O operations required for swapping pages in and out of memory as it is less likely to replace frequently used pages.

Disadvantages:

- Overhead: LRU requires overhead in terms of tracking page usage, which can be costly in terms of memory and processing resources.
- Complexity: Implementing LRU requires more complex algorithms, which can lead to increased system overhead and complexity.
- High startup cost: When the system starts up, LRU may initially perform poorly as the page history is not yet established, leading to more page faults.
- Increased cache thrashing: LRU can increase cache thrashing, where a page is repeatedly swapped in and out of memory, which can cause performance degradation.

Random

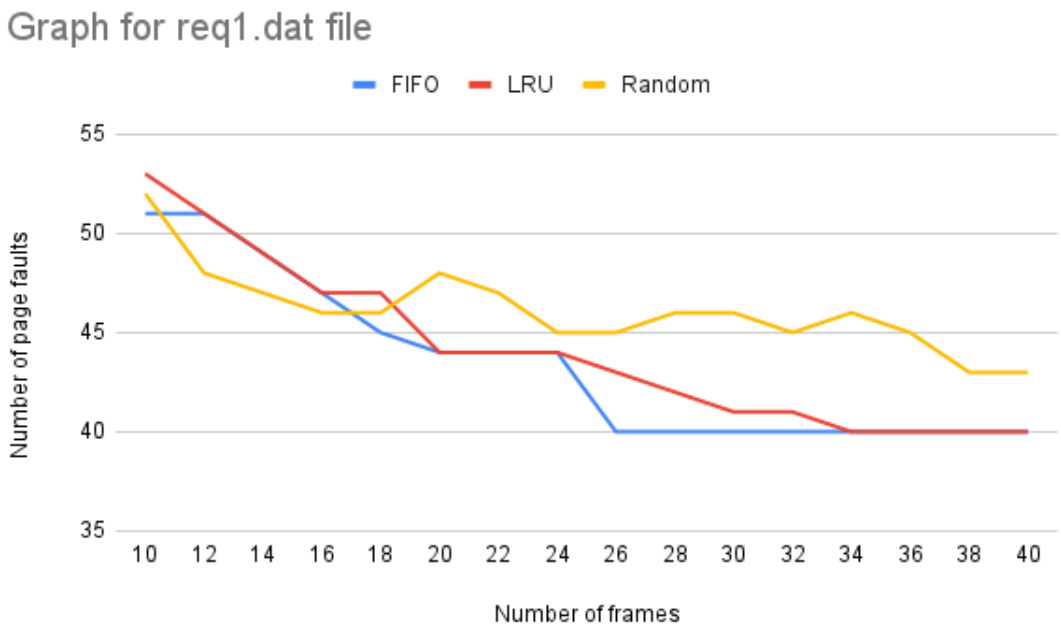
Advantages:

- Simple implementation: Random page replacement is simple to implement, as it doesn't require any tracking or history of page usage.
- No page history required: Random page replacement policy does not require any page history, which means it can handle new pages more efficiently.
- Fairness: Random page replacement provides fairness in terms of memory allocation to processes, as pages are selected randomly for replacement.

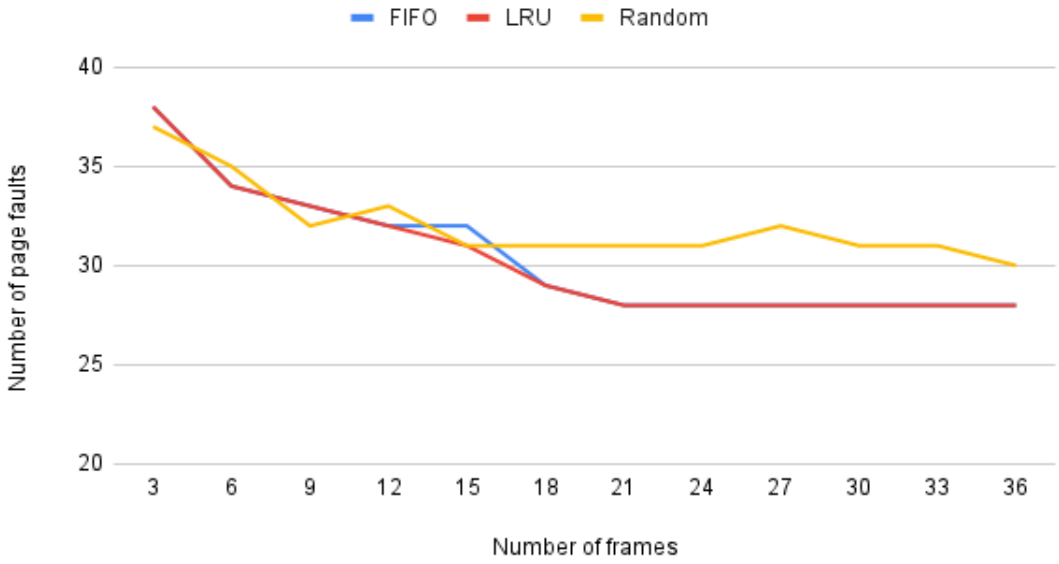
Disadvantages:

- Poor performance: Random page replacement policy has poor performance compared to other page replacement policies such as LRU, as it doesn't take into account the frequency of page usage.
- Inefficient use of memory: Random page replacement policy can lead to inefficient use of memory, as frequently used pages may get replaced, resulting in more page faults.
- Increased I/O overhead: Random page replacement policy can lead to increased I/O overhead, as frequently used pages may need to be swapped in and out of memory, resulting in more disk operations.
- Unpredictable behavior: Random page replacement policy is unpredictable in nature, as there is no way to predict which page will be selected for replacement.

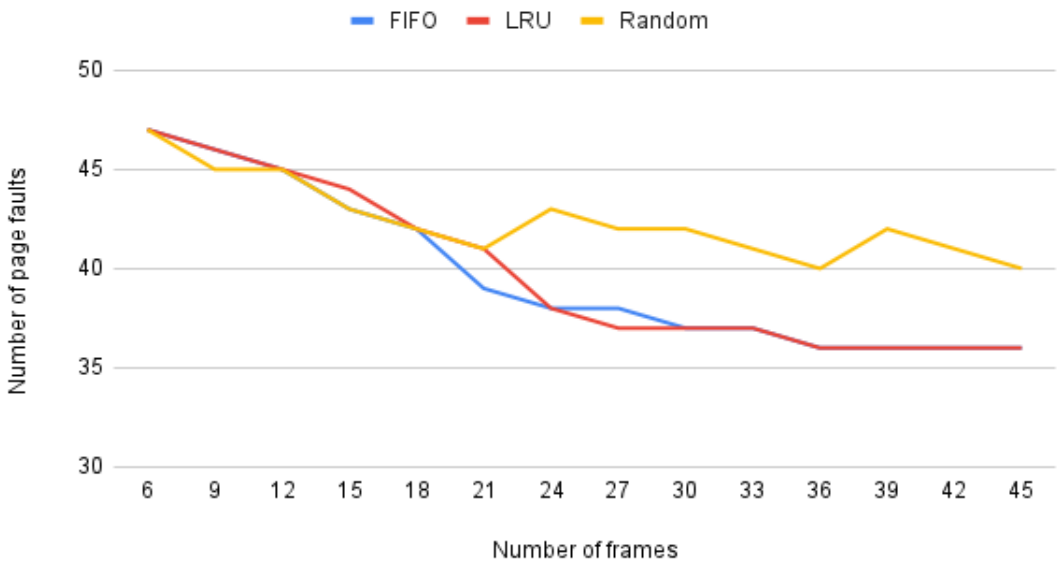
Graphs



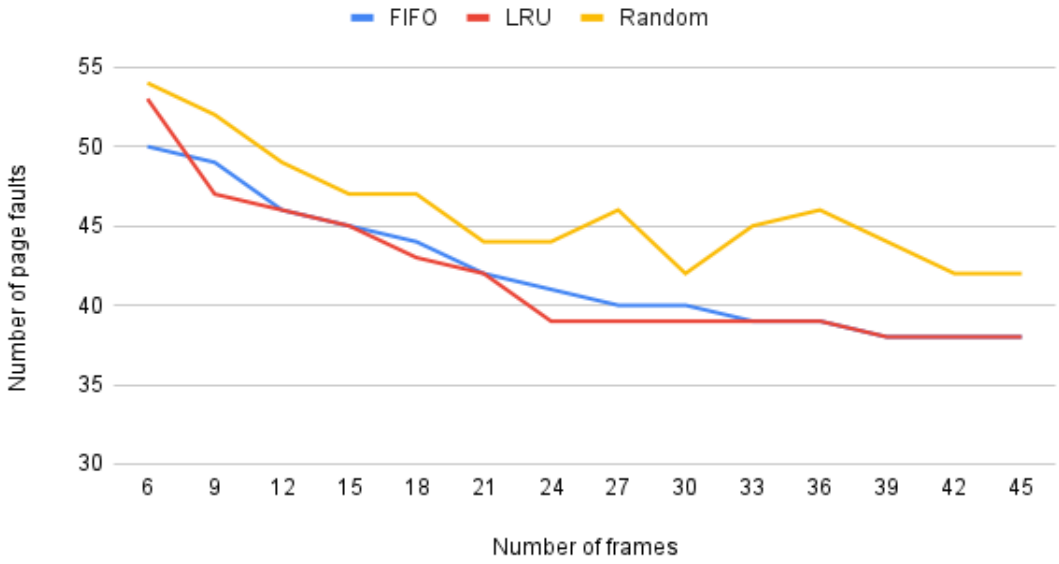
Graph for req2.dat file



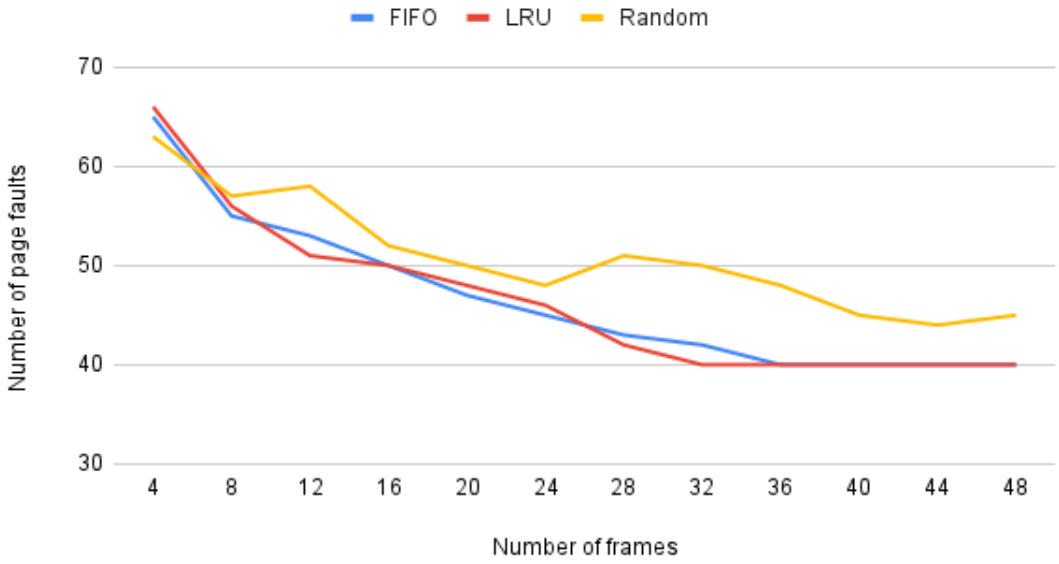
Graph for req3.dat file



Graph for req4.dat file



Graph for req5.dat file



Analysis

- The general trend that is observed is for fixed number of addressable pages and disk blocks for swap space as we increase the frame number, number of page faults decrease.
- We can't say anything for sure about the random page replacement policy but generally for fixed number of pages and blocks, page faults decrease as increase in frame number. Sometimes the number of page faults may even increase in between.
- In my test_case files, Belady's anomaly, which means that adding more memory may lead to more page faults, is not observed but it may occur with diffrent test_case files.
- The graphs demonstrate the effectiveness of the LRU page replacement policy over the FIFO policy in this scenario. However, the performance of page replacement policies depends on the specific access pattern of each process, and the results may differ in other scenarios.