
CS 314 – Operating Systems Lab

Lab7 Report

Name: Sahaja Nandyala

Roll-No: 200010032

1 Replacement policies

1.1 First In First Out (FIFO)

In First in First Out replacement policy when a page fault occurs and the physical memory is full, the oldest page i.e., the page that was brought into memory first is replaced. FIFO is simple policy but there it may be susceptible to thrashing.

To implement this a vector is used and when a new page comes it is pushed at back of vector and when a page fault occurs and physical memory is full then the first page in vector is removed to create space for new page.

1.2 Least Recently Used (LRU)

In Least Recently Used replacement policy when a page fault occurs and the physical memory is full, the page which is least recently used is removed to create space for the new page. When sequence of page request has locality LRU works well than FIFO. LRU has good balance between performance and complexity.

To implement this a vector is taken to maintain pages in physical memory and when a new page comes it is pushed at the back of vector and if an already existing page is requested then it is moved to the back of vector, if physical memory is full then the page at beginning of queue is removed to make space for the requested page.

1.3 Random

In Random replacement policy when physical memory is full, a page is randomly removed to make space for new page.

To implement this a vector is taken to maintain pages in physical memory and when a new page comes it is pushed at the back of vector, if physical memory is full then a page is selected randomly and removed to make space for new page.

2 Results and Observations

2.1 req1.dat

The page request sequence for req1.dat is as follows

1 10 32 2 12 5 2 16 9 30 21 35 6 8 7 17 22 38 45 53 43 10 8 20 30
16 18 56 60 57 53 27 35 24 32 13 17 4 5 18 20 52 28 25 18 9 19 3 31
59 11 6 23 28 37 48

Following table represents number of the page faults when number of addressable pages is set to 60 and number of blocks in swap space is set to 50 and number of pages in main memory are varying for FIFO, LRU, Random respectively for req1.dat

# Page frames	FIFO	LRU	Random
10	51	53	50
20	44	44	41
30	40	41	41
40	40	40	40
50	40	40	40

Graph for above tabulated results

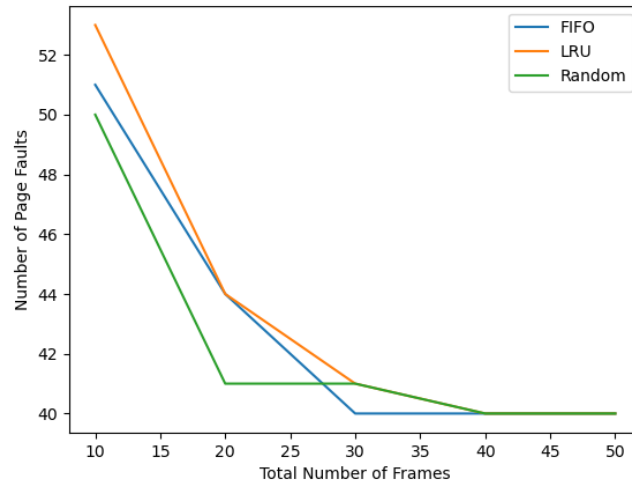


Figure 1: req1.dat

From the results we can see that FIFO is working better than LRU. As the number of frames in main memory increases replacement policy doesn't affect the efficiency in this case after 40 all replacement policies are giving same results. This sequence is an example where FIFO works better than LRU.

2.2 req2.dat

The page request sequence for req2.dat is as follows

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 1 2 3 4 5 6 7 8 9 10 11
12 13 14 15 16 17 18 19 20 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

Following table represents number of the page faults when number of addressable pages is set to 60 and number of blocks in swap space is set to 50 and number of pages in main memory are varying for FIFO, LRU, Random respectively for req2.dat

# Page frames	FIFO	LRU	Random
10	60	60	52
20	20	20	20
30	20	20	20
40	20	20	20
50	20	20	20

Graph for above tabulated results

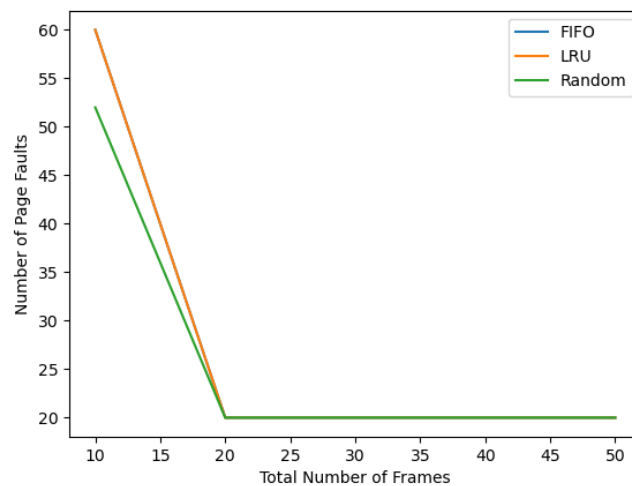


Figure 2: req2.dat

This workload is an example for looping sequential. Under a looping sequential workload, FIFO and LRU kick out older pages. Because of the workload's cyclical nature, these older pages will be viewed more quickly than the pages that the policies wish to preserve in cache. Here we can see that FIFO and LRU page faults decreases linearly as the number of frames are increasing.

2.3 req3.dat

The page request sequence for req3.dat is as follows

55 54 55 59 55 51 48 50 50 6 57 49 43 49 58 49 57 57 52 22 58 54 57 52 49
56 52 49 46 58 27 4 43 56 51 57 52 52 53 55 52 52 50 50 45 15 51 58 58 56

Following table represents number of the page faults when number of addressable pages is set to 60 and number of blocks in swap space is set to 50 and number of pages in main memory are varying for FIFO, LRU, Random respectively for req3.dat

# Page frames	FIFO	LRU	Random
10	27	26	28
20	20	20	20
30	20	20	20
40	20	20	20
50	20	20	20

Graph for above tabulated results

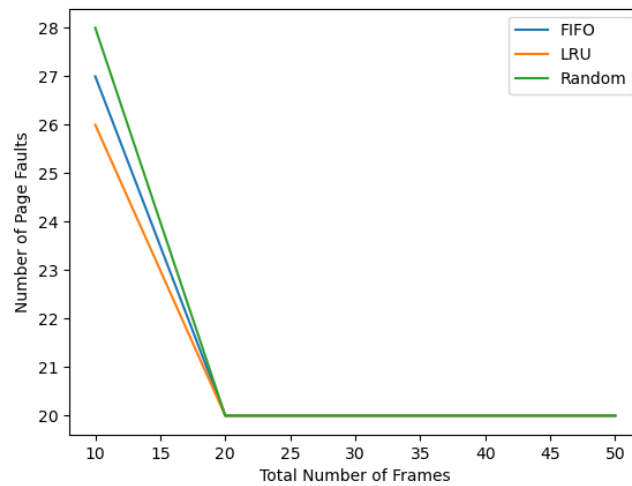


Figure 3: req3.dat

This sequence is similar to 80-20 workload which exhibits locality. In this can LRU gives good results than FIFO which we can see from the above results. And after a particular point replacement policy doesn't affect in this case after 20 LRU, FIFO, Random all are giving same results.

2.4 req4.dat

The page request sequence for req4.dat is as follows

16 33 22 58 36 18 39 1 38 42 54 9 57 34 6 49 21 34 36 44 59 19 46 47
58 20 6 47 44 33 45 51 52 9 17 39 23 8 39 19 27 3 19 2 41 42 33 28 15 37

Following table represents number of the page faults when number of addressable pages is set to 60 and number of blocks in swap space is set to 50 and number of pages in main memory are varying for FIFO, LRU, Random respectively for req4.dat

# Page frames	FIFO	LRU	Random
10	45	45	44
20	39	39	39
30	37	36	36
40	36	36	36
50	36	36	36

Graph for above tabulated results

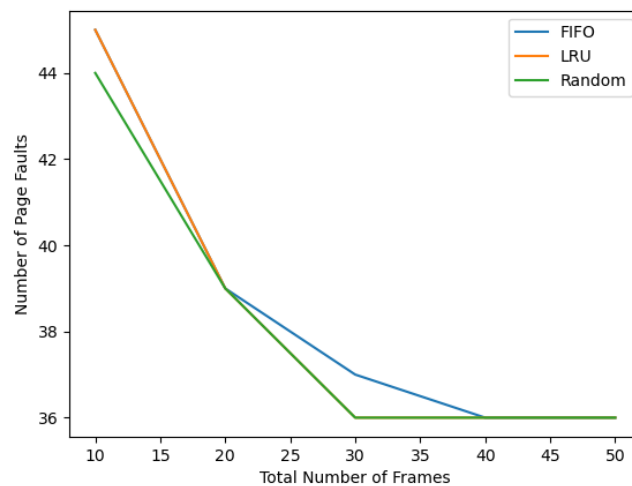


Figure 4: req4.dat

This sequence is generated randomly. we can see from the above graph that Random replacement policy is giving much better results than FIFO and LRU. We can see that as number of frames increases the page faults are decreasing.

2.5 req5.dat

The page request sequence for req5.dat is as follows

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 1 2 3 4 5 6 7 8 9 10 11 12 13 16
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Following table represents number of the page faults when number of addressable pages is set to 60 and number of blocks in swap space is set to 50 and number of pages in main memory are varying for FIFO, LRU, Random respectively for req5.dat

# Page frames	FIFO	LRU	Random
10	45	45	33
14	31	32	16
15	32	19	16
20	16	16	16
30	16	16	16

Graph for above tabulated results

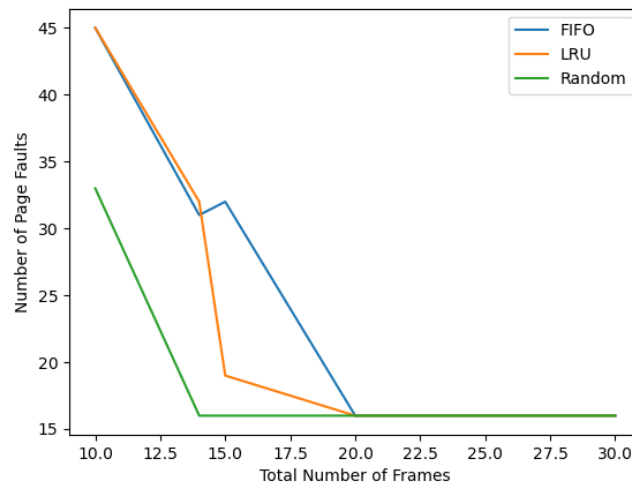


Figure 5: req5.dat

This sequence is an example for Be-lady's anomaly which states that at a point on as number of frames increases number of page faults for FIFO replacement policy. For this sequence we can observe spike in page faults from 14 page frames to 15 page frames.