운영체제[2021-2학기] Project-02 [Virtual Memory Management 기법 구현]

SWE3004\_41

2015310464 원종원

1. 가정

가정1. Time은 정수값으로 표현하며, 1부터 시작한다.

가정2.

가정3.

가정4.

2. 설계 및 구현에 사용한 도구

-사용언어 : C

-IDE : vscode

- Virtual Memory Management 설계 배경지식 : 운영체제 OS9~OS10 수업 참고

-특이사항 : 기본적으로 동일한 input에 대해 각각의 기법마다(총5번) Time 1~end 까지 실행 및 결과기록을 하도록 구성하였습니다.

MIN 기법과 LFU기법에서는 여러 page들간에 tie가 발생할 수 있기 때문에

이경우 tie-breaking을 일종의 random한 방법으로서 page frame번호가 작은 것을 victim으로 선정하도록 하였습니다.

MIN기법에서 tie가 발생하는 경우는 여러 페이지가 동일하게 무한대의 값을 가지는 경우입니다. 따라서 이 경우 여러 페이지 중 어떤 것을 선택하더라도 page fault수는 달라지지 않게 됩니다.

LFU기법에서 tie가 발생하는 경우는 여러 페이지가 메모리에 동일한 횟수만큼 올라왔을 경우입니다. 이경우에 LRU 기법을 적용하는 것을 고려해 보았으나, LFU기법을 사용하는 오버헤드와 LRU기법을 사용하는 오버헤드가 더해지면 현실적으로 성능이 크게 저하될 것이라고 생각하였습니다. 또한 이미 LFU기법을 통해 선정된 몇 개 안되는 victim 후보들 간에 선정하는 것이기 때문에 LRU 기법을 사용하더라도 극적인 page fault 감소는 없을 것이라고 판단하였습니다. 따라서 차라리 victim을 random하게 선정하여 오버헤드를 줄이는 것이 낫다고 판단하였습니다.

Working Set 기법에서 memory 내 모든 page frame이 full인 상황이라면, LRU기법으로 victim을 선정하도록 하였습니다. Working Set 기법 같은 경우 (page가 full인 상황에서) 메모리에서 빠져나가는 page가 없는 경우 반드시 replace가 이루어져야만 합니다. 그런데 이 경우에도 random하게 victim을 선정하게 된다면, page fault 수를 크게 늘리게 될 것이라고 판단하였습니다. 그 이유는, 만약 window size가 매우 크다고 가정하면, 매 time마다 memory에서 빠져나가는 페이지가 별로 없기 떄문에 높은 확률로 replace가 발생하게 됩니다. 따라서 이 경우에는 오버헤드를 감수하더라도 LRU 기법을 적용하여 page fault 수를 줄이는 것이 성능에 더 도움이 될것이라 판단하였습니다.

3. input에 대한 설명

교안을 참고하여 동일한 방식으로 input을 구성하였습니다.

7 4 3 14

1 2 6 1 4 5 1 2 1 4 5 6 4 5

4. output에 대한 설명

기존에 교안에서 보았던 Chart와 달리 종적으로 볼 수 있도록 구성하였습니다.

첫번째 줄에는 time과 reference string을 표시했습니다.

두번째 줄에는 page fault 여부를 표시하였는데, page fault인 경우에는 “Page Fault”로, page fault가 아닌 경우에는 “not in Page Fault”로 표시하였습니다.

Working Set 기법인 경우에, 해당 time에서 Working Set으로 선정된 page가 어떤 것인지 확인하기 위해 Working Set을 표시했습니다.

그다음 줄부터는 M개 만큼의 page frame 상태를 표시했습니다. 이때, page fault가 발생하여 load 혹은 replace가 이루어진 위치의 page frame에는 화살표를 나타냈습니다.

Reference String을 모두 처리하여 끝날 때는, page fault가 발생한 횟수를 표시하였고, page fault rate 또한 소수점 둘째자리까지 표시하였습니다. 이때 Working Set 기법의 경우에는 가변 할당 방식이기에 page fault rate가 아닌 평균적으로 할당된 frame 수를 소수점 둘째자리까지 표시했습니다.

5. 실행 결과를 보는 방법

첨부한 input.txt 파일을 명령인수로 넣어 OS41\_2021-2\_2015310464\_원종원\_P2.c.c 파일을 실행하면 됩니다.

6. 실행 결과

------------------Algorithm: MIN------------------

Time : 1 / Ref. string : 1

Page Fault

<Memory state>

page frame 1 : 1 <-

page frame 2 : -

page frame 3 : -

page frame 4 : -

Time : 2 / Ref. string : 2

Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2 <-

page frame 3 : -

page frame 4 : -

Time : 3 / Ref. string : 6

Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2

page frame 3 : 6 <-

page frame 4 : -

Time : 4 / Ref. string : 1

not in Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2

page frame 3 : 6

page frame 4 : -

Time : 5 / Ref. string : 4

Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2

page frame 3 : 6

page frame 4 : 4 <-

Time : 6 / Ref. string : 5

Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2

page frame 3 : 5 <-

page frame 4 : 4

Time : 7 / Ref. string : 1

not in Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2

page frame 3 : 5

page frame 4 : 4

Time : 8 / Ref. string : 2

not in Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2

page frame 3 : 5

page frame 4 : 4

Time : 9 / Ref. string : 1

not in Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2

page frame 3 : 5

page frame 4 : 4

Time : 10 / Ref. string : 4

not in Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2

page frame 3 : 5

page frame 4 : 4

Time : 11 / Ref. string : 5

not in Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2

page frame 3 : 5

page frame 4 : 4

Time : 12 / Ref. string : 6

Page Fault

<Memory state>

page frame 1 : 6 <-

page frame 2 : 2

page frame 3 : 5

page frame 4 : 4

Time : 13 / Ref. string : 4

not in Page Fault

<Memory state>

page frame 1 : 6

page frame 2 : 2

page frame 3 : 5

page frame 4 : 4

Time : 14 / Ref. string : 5

not in Page Fault

<Memory state>

page frame 1 : 6

page frame 2 : 2

page frame 3 : 5

page frame 4 : 4

the number of Page Fault : 6

Page Fault Rate : 0.43

------------------Algorithm: FIFO------------------

Time : 1 / Ref. string : 1

Page Fault

<Memory state>

page frame 1 : 1 <-

page frame 2 : -

page frame 3 : -

page frame 4 : -

Time : 2 / Ref. string : 2

Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2 <-

page frame 3 : -

page frame 4 : -

Time : 3 / Ref. string : 6

Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2

page frame 3 : 6 <-

page frame 4 : -

Time : 4 / Ref. string : 1

not in Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2

page frame 3 : 6

page frame 4 : -

Time : 5 / Ref. string : 4

Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2

page frame 3 : 6

page frame 4 : 4 <-

Time : 6 / Ref. string : 5

Page Fault

<Memory state>

page frame 1 : 5 <-

page frame 2 : 2

page frame 3 : 6

page frame 4 : 4

Time : 7 / Ref. string : 1

Page Fault

<Memory state>

page frame 1 : 5

page frame 2 : 1 <-

page frame 3 : 6

page frame 4 : 4

Time : 8 / Ref. string : 2

Page Fault

<Memory state>

page frame 1 : 5

page frame 2 : 1

page frame 3 : 2 <-

page frame 4 : 4

Time : 9 / Ref. string : 1

not in Page Fault

<Memory state>

page frame 1 : 5

page frame 2 : 1

page frame 3 : 2

page frame 4 : 4

Time : 10 / Ref. string : 4

not in Page Fault

<Memory state>

page frame 1 : 5

page frame 2 : 1

page frame 3 : 2

page frame 4 : 4

Time : 11 / Ref. string : 5

not in Page Fault

<Memory state>

page frame 1 : 5

page frame 2 : 1

page frame 3 : 2

page frame 4 : 4

Time : 12 / Ref. string : 6

Page Fault

<Memory state>

page frame 1 : 5

page frame 2 : 1

page frame 3 : 2

page frame 4 : 6 <-

Time : 13 / Ref. string : 4

Page Fault

<Memory state>

page frame 1 : 4 <-

page frame 2 : 1

page frame 3 : 2

page frame 4 : 6

Time : 14 / Ref. string : 5

Page Fault

<Memory state>

page frame 1 : 4

page frame 2 : 5 <-

page frame 3 : 2

page frame 4 : 6

the number of Page Fault : 10

Page Fault Rate : 0.71

------------------Algorithm: LRU------------------

Time : 1 / Ref. string : 1

Page Fault

<Memory state>

page frame 1 : 1 <-

page frame 2 : -

page frame 3 : -

page frame 4 : -

Time : 2 / Ref. string : 2

Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2 <-

page frame 3 : -

page frame 4 : -

Time : 3 / Ref. string : 6

Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2

page frame 3 : 6 <-

page frame 4 : -

Time : 4 / Ref. string : 1

not in Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2

page frame 3 : 6

page frame 4 : -

Time : 5 / Ref. string : 4

Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2

page frame 3 : 6

page frame 4 : 4 <-

Time : 6 / Ref. string : 5

Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 5 <-

page frame 3 : 6

page frame 4 : 4

Time : 7 / Ref. string : 1

not in Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 5

page frame 3 : 6

page frame 4 : 4

Time : 8 / Ref. string : 2

Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 5

page frame 3 : 2 <-

page frame 4 : 4

Time : 9 / Ref. string : 1

not in Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 5

page frame 3 : 2

page frame 4 : 4

Time : 10 / Ref. string : 4

not in Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 5

page frame 3 : 2

page frame 4 : 4

Time : 11 / Ref. string : 5

not in Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 5

page frame 3 : 2

page frame 4 : 4

Time : 12 / Ref. string : 6

Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 5

page frame 3 : 6 <-

page frame 4 : 4

Time : 13 / Ref. string : 4

not in Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 5

page frame 3 : 6

page frame 4 : 4

Time : 14 / Ref. string : 5

not in Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 5

page frame 3 : 6

page frame 4 : 4

the number of Page Fault : 7

Page Fault Rate : 0.50

------------------Algorithm: LFU------------------

Time : 1 / Ref. string : 1

Page Fault

<Memory state>

page frame 1 : 1 <-

page frame 2 : -

page frame 3 : -

page frame 4 : -

Time : 2 / Ref. string : 2

Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2 <-

page frame 3 : -

page frame 4 : -

Time : 3 / Ref. string : 6

Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2

page frame 3 : 6 <-

page frame 4 : -

Time : 4 / Ref. string : 1

not in Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2

page frame 3 : 6

page frame 4 : -

Time : 5 / Ref. string : 4

Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2

page frame 3 : 6

page frame 4 : 4 <-

Time : 6 / Ref. string : 5

Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 5 <-

page frame 3 : 6

page frame 4 : 4

Time : 7 / Ref. string : 1

not in Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 5

page frame 3 : 6

page frame 4 : 4

Time : 8 / Ref. string : 2

Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2 <-

page frame 3 : 6

page frame 4 : 4

Time : 9 / Ref. string : 1

not in Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2

page frame 3 : 6

page frame 4 : 4

Time : 10 / Ref. string : 4

not in Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2

page frame 3 : 6

page frame 4 : 4

Time : 11 / Ref. string : 5

Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 2

page frame 3 : 5 <-

page frame 4 : 4

Time : 12 / Ref. string : 6

Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 6 <-

page frame 3 : 5

page frame 4 : 4

Time : 13 / Ref. string : 4

not in Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 6

page frame 3 : 5

page frame 4 : 4

Time : 14 / Ref. string : 5

not in Page Fault

<Memory state>

page frame 1 : 1

page frame 2 : 6

page frame 3 : 5

page frame 4 : 4

the number of Page Fault : 8

Page Fault Rate : 0.57

------------------Algorithm: WS------------------

Time : 1 / Ref. string : 1

Page Fault

Working Set : { 1 }

<Memory state>

page frame 1 : 1 <-

page frame 2 : -

page frame 3 : -

page frame 4 : -

Time : 2 / Ref. string : 2

Page Fault

Working Set : { 1 2 }

<Memory state>

page frame 1 : 1

page frame 2 : 2 <-

page frame 3 : -

page frame 4 : -

Time : 3 / Ref. string : 6

Page Fault

Working Set : { 1 2 6 }

<Memory state>

page frame 1 : 1

page frame 2 : 2

page frame 3 : 6 <-

page frame 4 : -

Time : 4 / Ref. string : 1

not in Page Fault

Working Set : { 1 2 6 }

<Memory state>

page frame 1 : 1

page frame 2 : 2

page frame 3 : 6

page frame 4 : -

Time : 5 / Ref. string : 4

Page Fault

Working Set : { 1 2 4 6 }

<Memory state>

page frame 1 : 1

page frame 2 : 2

page frame 3 : 6

page frame 4 : 4 <-

Time : 6 / Ref. string : 5

Page Fault

Working Set : { 1 4 5 6 }

<Memory state>

page frame 1 : 1

page frame 2 : 5 <-

page frame 3 : 6

page frame 4 : 4

Time : 7 / Ref. string : 1

not in Page Fault

Working Set : { 1 4 5 }

<Memory state>

page frame 1 : 1

page frame 2 : 5

page frame 3 : -

page frame 4 : 4

Time : 8 / Ref. string : 2

Page Fault

Working Set : { 1 2 4 5 }

<Memory state>

page frame 1 : 1

page frame 2 : 5

page frame 3 : 2 <-

page frame 4 : 4

Time : 9 / Ref. string : 1

not in Page Fault

Working Set : { 1 2 5 }

<Memory state>

page frame 1 : 1

page frame 2 : 5

page frame 3 : 2

page frame 4 : -

Time : 10 / Ref. string : 4

Page Fault

Working Set : { 1 2 4 }

<Memory state>

page frame 1 : 1

page frame 2 : 4 <-

page frame 3 : 2

page frame 4 : -

Time : 11 / Ref. string : 5

Page Fault

Working Set : { 1 2 4 5 }

<Memory state>

page frame 1 : 1

page frame 2 : 4

page frame 3 : 2

page frame 4 : 5 <-

Time : 12 / Ref. string : 6

Page Fault

Working Set : { 1 4 5 6 }

<Memory state>

page frame 1 : 1

page frame 2 : 4

page frame 3 : 6 <-

page frame 4 : 5

Time : 13 / Ref. string : 4

not in Page Fault

Working Set : { 4 5 6 }

<Memory state>

page frame 1 : -

page frame 2 : 4

page frame 3 : 6

page frame 4 : 5

Time : 14 / Ref. string : 5

not in Page Fault

Working Set : { 4 5 6 }

<Memory state>

page frame 1 : -

page frame 2 : 4

page frame 3 : 6

page frame 4 : 5

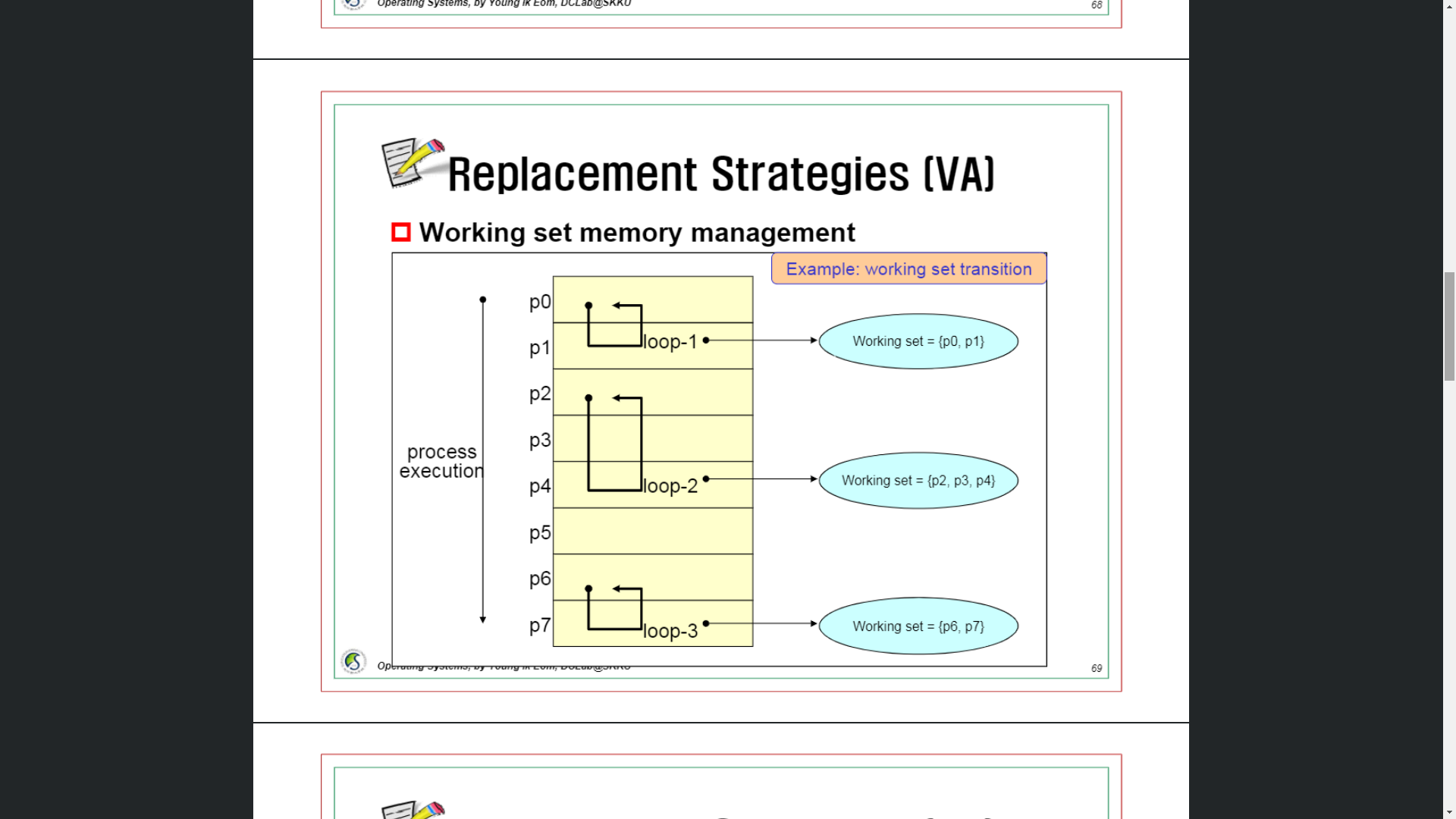
7. 결과 설명(performance 측면에서)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | MIN | FIFO | LRU | LFU | WS  (window:3) | WS  (window:5) |
| Page Faults | 6 | 10 | 7 | 8 | 9 | 7 |
| Page Fault rate | 0.43 | 0.71 | 0.5 | 0.57 | - | - |
| Avg. allocated frames | 4 | 4 | 4 | 4 | 3.14 | 3.5 |
| Page Faults with same number of allocated frames | 6 | 10 | 7 | 8 | 11.4 | 8 |

WS 기법은 가변 할당 방식이기 때문에 일반적인 page fault rate로 나머지 네가지 방법과 성능을 비교할 수 없었습니다. 따라서 WS 기법에서 평균 할당 프레임이 나머지 네 기법과 같다고 가정하고(Avg. allocated frames=4) page fault 수를 조정했을 때의 성능을 비교했습니다. 즉 page fault개수를 줄이는 측면을 놓고 평가했을 때, 역시나 숫자가 작을수록 좋은 성능을 내는 방법이라는 것을 알 수 있습니다.

MIN>LRU>LFU>FIFO>WS 순으로 좋은 성능을 내는 것을 확인할 수 있었습니다. 이론적으로 MIN이 page fault 수를 가장 낮출 수 있는 방법이기 때문에 월등한 성능을 보여주었지만, WS의 경우 매우 낮은 성능을 보였습니다. 저는 이 결과를 window size가 작기 때문에 working set에 들어왔던 페이지를 너무 성급하게 내보냈던 것으로 해석하였고, 따라서 window size를 3에서 5로 늘려 다시 실행해 보았습니다. 그 결과 WS 기법이 LFU 기법 수준으로 성능이 올라간 것을 확인할 수 있었습니다.

즉 WS기법에서는 window size가 달라짐에 따라 성능이 달라진다는 것을 확인할 수 있었고, 최적의 성능을 위해서는 프로세스의 page 참조 패턴에 따라서 window size를 조정함으로써 memory management를 최적화 할 수 있겠다고 생각했습니다.



즉 어떤 프로세스가 다음과 같이 여러 페이지들을 loop을 돌면서 참조한다고 했을 때, 상대적으로 긴 loop를 많이 도는 프로세스에게는 window size를 크게 하고, 상대적으로 짧은 loop을 많이 도는 프로세스에게는 window size를 작게 잡는 것이 page fault를 낮추고, memory 점유를 낮추는데 도움이 된다는 결론을 내렸습니다..