

EOPSY

Laboratory 4

Bartłomiej Mastej
album number: 300172

May 4, 2021

1 Execution

In order to map any 8 pages of physical memory to the first 8 pages of the virtual memory there was a need to make changes in the file *memory.conf* - there was mapped for each first of the virtual memory addresses exactly one random physical memory page. Further, so as to read from one virtual memory address on each of the 64 virtual pages the following operations were performed: firstly, the page size was checked to be equal 16384. Secondly, for each next page, the value of the page was multiplied by consecutive number and then 1 was added. As a result, each of 64 virtual memory pages had successfully read exactly one virtual memory address.

2 Observations

While running in the single step mode the following things were observed:

1. The mapping of the first 8 pages of physical memory to first 8 pages of the virtual memory was done correctly. That can be seen in the appendix.
2. For the next pages, up to page number 31, there is done the mapping such that for each virtual page there is mapped the physical page of the same number as the virtual one, despite the fact that it was removed from the configuration file.
3. Since the virtual page number 32 there is a page fault spotted for each next virtual page.
4. One can see that the pages are iterated in the order of the Virtual Memory Page.
5. As the fault is spotted since the 32 virtual page then the mapping is repeated the same as it was done for the first 32 pages for example virtual page 32 points to the physical page 11 - the same assignment as for the virtual page 0.

3 Conclusions & Thoughts

What had been expected to be faulted was when there was the first time encountered the mapping of the same physical page to two different virtual pages, nevertheless, it did not occur.

As the order of the execution is the same as the page order, thus one can observed that FIFO algorithm was used for sure. Moreover, there is a field *Pagefault : YES/NO* which might be used in the *SecondChanceAlgorithm*. As it was observed each page which is faulting gets the physical page assignment of the first, in the sense of order, virtual page. All in all, it is obvious that the algorithm that was used must be the *SecondChanceReplacementAlgorithm*.

4 Appendix

The output of the tracefile can be seen below:

```
READ 0 ... okay
READ 4000 ... okay
READ 8000 ... okay
READ c000 ... okay
READ 10000 ... okay
READ 14000 ... okay
READ 18000 ... okay
READ 1c000 ... okay
READ 20000 ... okay
READ 24000 ... okay
READ 28000 ... okay
READ 2c000 ... okay
READ 30000 ... okay
READ 34000 ... okay
READ 38000 ... okay
READ 3c000 ... okay
READ 40000 ... okay
READ 44000 ... okay
READ 48000 ... okay
READ 4c000 ... okay
READ 50000 ... okay
READ 54000 ... okay
READ 58000 ... okay
READ 5c000 ... okay
READ 60000 ... okay
READ 64000 ... okay
READ 68000 ... okay
READ 6c000 ... okay
READ 70000 ... okay
READ 74000 ... okay
```

READ 78000 ... okay
READ 7c000 ... okay
READ 80000 ... page fault
READ 84000 ... page fault
READ 88000 ... page fault
READ 8c000 ... page fault
READ 90000 ... page fault
READ 94000 ... page fault
READ 98000 ... page fault
READ 9c000 ... page fault
READ a0000 ... page fault
READ a4000 ... page fault
READ a8000 ... page fault
READ ac000 ... page fault
READ b0000 ... page fault
READ b4000 ... page fault
READ b8000 ... page fault
READ bc000 ... page fault
READ c0000 ... page fault
READ c4000 ... page fault
READ c8000 ... page fault
READ cc000 ... page fault
READ d0000 ... page fault
READ d4000 ... page fault
READ d8000 ... page fault
READ dc000 ... page fault
READ e0000 ... page fault
READ e4000 ... page fault
READ e8000 ... page fault
READ ec000 ... page fault
READ f0000 ... page fault
READ f4000 ... page fault
READ f8000 ... page fault
READ fc000 ... page fault