EOPSY Lab 4- Memory management

Jakub Prochorec

In this excercise, the goal was to use a simulator MemoryManagement written in Java which based on configurations set in “commands” and “memory.conf” files shows step-by-step the work of page replacement algorithm. There were 64 virtual pages and 32 physical pages. The first 8 were set by me in memory.conf file, while the rest (24) was set by default by the program. In the commands file, I wrote 64 operations reflecting 64 reads from corresponding 64 virtual pages ( in a fashion read hex 0; read hex 4000;….; read hex F8000;read hex FC000). I’ve run simulation one step at the time, and the final result was stored in a “tracefile” file. The results were the following : out of the total of 64 reads, 32 returned okay status, while 32 caused page fault. A page fault occurs when a program attempts to access a block of memory that is not stored in the physical memory, or RAM. The fault notifies the operating system that it must locate the data in virtual memory, then transfer it from the storage device, such as an HDD or SSD, to the system RAM. So in this case, there were in total 32 physical pages mapped to 32 virtual pages, and 32 of virtual pages were not mapped. So obviously if the attempt of read was done on the virtual page which had corresponding physical page, the returned status was okay as it could be immediately accessed. On the other hand, if the virtual page had no corresponding physical page, then page fault occurred because the operating system in this case had to transfer that data from virtual memory to physical one and use a page replacement algorithm. The page replacement which I observed was LRU (Least Recently Used). LRU works on the idea that pages that have been most heavily used in the past few instructions are most likely to be used heavily in the next few instructions too. So what was happening was that the first page was being replaced, then the second, etc. This algorithm in simple cases is actually very similar to FIFO, so it’s hard to judge which one exactly was that in action.

Summing up, I’ve made a simulation based on the requirements of the task description in regard to page assignment and read operations, and I have stepped through the simulation as well as analysed the output file to observe how the memory management was handled, when do the page faults occur and what kind of page replacement algorithm was being used.