

CSE 306 – Operating Systems
Project 2: Memory Management

Name: Snigdha Kamal

Student ID: 110937472

I pledge my honor that all parts of this project were done by me individually and without collaboration
with anybody else.

Statistics for Demo.jar

Number of Pages Swapped in : 1507

Number of Pages Swapped Out: 359

| Snapshot | CPU Utilization | Service Time per thread | Normalized Service time per thread |
|----------|-----------------|-------------------------|------------------------------------|
| 1 | 98.979996% | 16663.87 | 0.09238082 |
| 2 | 76.058% | 21099.115 | 0.04946768 |
| 3 | 62.850666% | 29337.541 | 0.046074647 |
| 4 | 66.898% | 23534.7 | 0.06311941 |
| 5 | 73.2024% | 23404.96 | 0.061316643 |
| 6 | 73.794% | 21437.959 | 0.065844126 |
| 7 | 72.78% | 24805.309 | 0.059149228 |
| 8 | 74.3085% | 24030.326 | 0.058917653 |
| 9 | 73.26667% | 26143.55 | 0.057164606 |
| 10 | 73.0676% | 27591.291 | 0.057233144 |

Statistics for my OSP Program with LRU Page Replacement

Number of Pages Swapped In: 1087

Number of Pages Swapped Out: 538

| Snapshot | CPU Utilization | Service Time per Thread | Normalized Service Time per thread |
|----------|-----------------|-------------------------|------------------------------------|
| 1 | 98.743996% | 11832.6 | 0.075775914 |
| 2 | 99.372% | 19106.707 | 0.060425118 |
| 3 | 97.90133% | 19238.482 | 0.064085536 |
| 4 | 95.341995% | 22544.318 | 0.07301413 |
| 5 | 94.2288% | 22870.373 | 0.06705244 |
| 6 | 93.54867% | 23397.562 | 0.058735542 |
| 7 | 87.47372% | 26888.17 | 0.060615826 |
| 8 | 82.839% | 27451.504 | 0.06300273 |
| 9 | 81.763115% | 27634.85 | 0.062362816 |
| 10 | 82.4088% | 26704.455 | 0.06322917 |

Explanation of Results:

The number of pages swapped in is more in demo.jar because it only finds a replaceable frame as a victim frame without any special considerations. In the LRU scheme, the number of pages swapped in is low because it uses the least recently used frame as a victim frame. So in the first case, a highly used page could be replaced increasing the number of swap-ins but that won't happen in the LRU scheme. The number of Swap-outs in my scheme are greater because of the page cleaning daemon which is running in the background swapping 6 least recently used dirty pages out into the swap disk.

CPU utilization is higher in case of the LRU strategy compared to the Demo because of locality of reference. As LRU only swaps out the least recently used page, CPU can focus on processing and the number of page faults will be lesser compared to Demo, where the CPU will go into waiting often in order to service the page faults which will be higher in number compared to LRU. Hence, CPU utilization is better for LRU than for Demo.

Service time per thread and normalized service time remain same more or less for both strategies.