Page Replacement Algorithms – Results and Analysis

The purpose of this project was to explore differing levels of efficacy for three separate page replacement algorithms. Reduction of page faults was the goal. Based on our results, FIFO showed the most page faults, followed by LRU, and finally OPT showing the least amount of page faults. However, LRU and OPT exhibited similar trend. LRU may show slightly better performance than OPT for unique runs of the program. Both LRU and OPT are very similar stack-based algorithms. We believe this is related to the level of randomness achieved by the built-in Python method that was used to generate the page reference string on a per run basis. A more robust implementation of random number generator, combined with a much larger sample size of the random page reference string, may result in OPT consistently performing superior to both algorithms. However, in accordance with the assignment’s SOP, we kept to a sample size of 100 for our data collection.

A close up of a device

Description automatically generated

Therefore, the performance of the algorithms is as follows:

1st – Optimal (OPT)  
2nd – Least Recently Used (LRU)  
3rd – First In, First Out (FIFO)

These results align with the expected performance of each respective algorithm.