Adrian Gonzalez

Saba Jamalian

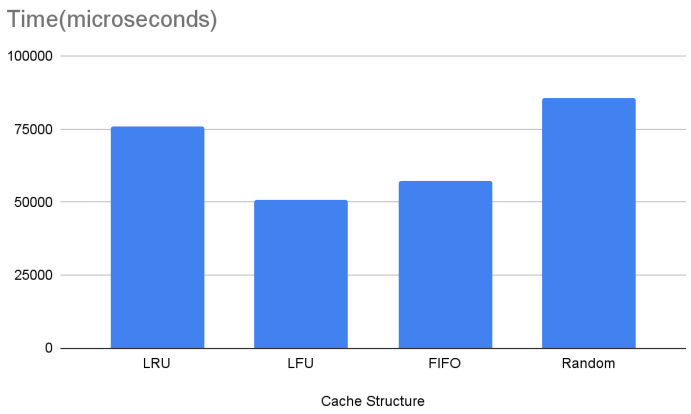
CS 210

May 13, 2025

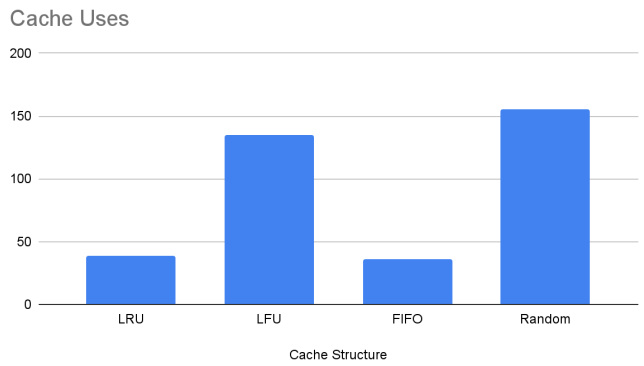
Analysis report

In this project, I was given world_cities.csv and constructed a project that could first read the csv file and allow the user to search through the file using country code and city name as input and storing the last 10 items into a cache. Then, the program added an option to use 4 different cache types; the first cache type implemented (Least Recently Used),a LFU (Least Frequently Used) cache, a FIFO (First In First Out) cache, and a random replacement cache. Afterwards, a trie data structure was used to load the csv file. Finally, I automated a script that would test all four cache types 1000 times by timing their duration and counting the times the cache was accessed.

1.) Duration for each cache type



2.) Total cache accesses



According to the data, the LFU cache looks like the best fit for this project with the lowest runtime. LRU looks like the weakest data type due to its higher runtime and cache accesses. FIFO doesn't look very efficient either, with a very low cache access and since it removes cached items based on order rather than frequency like LFU, it doesn't account for the popularity/frequency of certain searches. Random cache is also not as consistent as the other due to its random removals. Therefore, LFU cache is the best cache for this project according to the data and compared to the other cache types