For my cache project, the main data structures being used are arrays of structures. I made a cache structure that had properties of a cache as attributes such as cache_size int, block_size int, and cache policy char*. Cache structures also have an array of set structures as an attribute. I made each set have an array of lines structures. Finally, each line had a valid bit int attribute, a tagbits unsigned long attribute, and entryorder int for keeping track of the FIFO policy. I had intended to also work on LRU but did not manage to finish in time, thus each line also has a usedorder int attribute. The prefetcher changed the number of cache hits and memory reads because it is exploiting spatial locality more than a non-prefetcher cache each time a memory trace misses the cache. Instead of storing the block you missed on a cache miss, with the prefetcher you are also fetching an additional block that is spatially close to the block you put into the cache initially(the next block in memory to be precise).