We present a novel replacement policy for last-level caches (LLCs). The fundamental observation is to view LLCs as a shared resource among multiple address streams with each stream being generated by a static memory access instruction. The management of LLCs in both single-core and multi-core processors can then be modeled as a competition among multiple instructions. We prioritize those instructions based on the number of LLC accesses and reuses and only allow cache lines having high instruction priorities to replace those of low priorities. Our experimental results based on a set of SPEC 2006 benchmarks show that it achieves significant performance improvement upon the least-recently used (LRU) replacement policy for benchmarks with high numbers of LLC misses. To handle LRU-friendly workloads, the set sampling technique is adopted to retain the benefits from the LRU replacement policy.