Indian Institute of Technology Mandi
February-June 2017 Semester

CS202: Data Structure and Algorithms

Programming Assignment 4 Problem Statements

Last date of submission of code: 27th March, 2017

Implement one of the following problems using C++ programing language.
Note:

1. Write a separate main programs to evaluate the functions in Chained Hashing, Linear Probing and Double Hashing data structures. The main functions should have the options to read inputs from user and display the contents of hash.
2. Implement the linked list data structure using the Dictionary.hpp
3. Implement Chained Hash, Linear Probing and Double Hashing data structures using ChainedMap.hpp, OpenMap.hpp and DoubleHashMap.hpp.
4. Use sequential linear list data structure as per the need of the problem assigned to you.
5. Write a separate main programs to evaluate the LRY cache problem.

Problem:

Least Recently Used (LRU) cache: Cache hold M memory block while main memory hold N blocks ($M << N$). Each cache block and main memory block contain K words. Initially no blocks in cache. When processor requests for a word, a block containing that word will be moved from main memory to cache i.e. main memory block number containing that word is mapped onto to the cache memory. A block can be place anywhere in the cache and each cache location contain only one block. Such a cache is called *associative mapped cache*. When processor requests for a word at each time, cache has to check whether the requesting word (i.e. block containing that word) is there in cache (i.e. *cache hit*) or not (*cache miss*). If already present in cache, then no need to bring new block. If not present in cache, i.e., when cache miss occurs and the cache is not full, a block containing that word need to be brought to cache from main memory and place in the empty space. During that process, if the cache is full, then LRU scheme is used to remove the least recently used block so that newly referenced block can be placed in the memory. Write a program in C++ to simulate this process.
Cache memory can be seen as a hash table. Decide which hashing technique is suitable. Main memory block number is the key. Main memory is seen as consecutive address locations starting from zero. Each word occupy one address location in the main memory. Input to the program is sequence of word addresses. Use a sequential list (array) to monitor least recently used cache block. The maximum size of the list will be equal to the cache size (M). Each location in a list is associated with a cache location. When a cache block is referenced (either it is in cache or brought from main memory), a counter associated with that block is set to zero. At that time the counter associated with all other blocks is incremented by 1. A block with highest counter number is the least recently used.

Reference:
Chapter 5, Section 5.5.1, 5.5.2: Carl Hamacher, Zvonko Vranesic, Safwat Zaky, "Computer Organization", Fifth Edition