Computer Organization Project 4: Cache Architecture.

Using your preferred programming language, program a Cache memory simulator. The tool should do the following:

1. (25) Prompt the user to provide the nominal size of the cache in Bytes. Prompt the user to provide the number of words per block (1, 2, 4, 8). Prompt the user to enter the mapping policy: DM or Set Associative. In the case of Set Associative, prompt the user to provide the number of blocks per set (N in the N-way definition)
2. (25) Outputs: (*) the number of blocks in cache, (*) the number of sets, (*) the way the address is partitioned (offset, index, tag),(*) the real size of cache.
3. (25) Prompt the user to provide a word address and output: where the word is located in cache, and whether this access is a hit or a miss. Keep a table updated with all the accesses provided by the user. Include the option of clearing cache.
4. (25) Simulation mode: generate sets of word addresses to populate/access the cache memory. Show the hit/miss rate after the simulation.

   Extra points (10 each)
5. In simulation mode, include the option of providing locality, such that the addresses are not just random but respond to some pattern
6. Provide a graphical interface so the evolution of cache population can be seen in the screen
7. Implement LRU replacement policy.

Make a video recording a demo of your project, and how it does each of the things asked above. Give me the scoop of how you wrote the program. Show that it works with four different cache configurations at least: Direct Mapped- one word per block and multiple words per block, and Set associative- one word per block and multiple words per block. You will submit your code and video under this assignment.