Hints for using PIN to generate traces.

1. You may have to write your own PIN tool to generate cache filtered DRAM traces. Some examples for cache filtering may be found in source/Tools/Memory/allcache.c under PIN source code.
2. While generating traces, please remember to comment out any print statements. Pinting to stdout introduces much higher tracing overheads. One may use memory buffers (there are examples in source/Tools) to store the traces, and finally print at the end of the program (or dump out the trace at regular intervals).
3. You may use RDTSC instruction to collect the timestamps (again, there are some examples in source/Tools). Timestamps will be elongated due to instrumentation overheads. Running a pre-processing step might be required to estimate the actual timestamps. Students may normalise the timestamp using the total time of execution under PIN, and time for execution without PIN. That is, determine the elongation factor due to PIN overheads as the execution time without PIN / execution time with PIN.