Lab 1: Running Microarchitectural Simulation Using SimpleScalar

In this lab, you need to simulate sim-outorder simulator from simplescalar suite. You will need a unix system for this exercise. The simulator can be downloaded from here: http://www.simplescalar.com/. You need to download simplesim-3v0e.tgz file

The benchmarks can be downloaded from here:
http://faculty.cse.tamu.edu/djimenez/614-spring14/hw4/benchmarks/index.html

A student with roll number K has to download benchmark number (K%28)+1.
For example, if your roll number is 18116067, then, you have to download benchmark number 12, since (18116067%28)+1 = 12. This benchmark is health.eio.gz.

The L2 cache size, which a student has to use, has to be found like this:
Let p = int((RollNumber%112)/28)
Then, for each p, find the cache size as shown below.

p	cache size
0	512KB
1	1MB
2	2MB
3	4MB

For example, p = int((18116067%112)/28) =2, so cache size is 2MB.

Except the benchmark and cache size, remaining parameters are same for all students. In other words, each student needs to run benchmark with cache configuration specified for him/her (note that due to this, each student will get different results).

Run sim-outorder for 10,000,000 instructions with cache configuration given for you and the benchmark given to you. (Other than these, you don't need to change any parameter). L2 cache block size is 64B and L2 associativity is 8 throughout this exercise.

(a) Find the L2 cache miss-rate (=L2_misses/L2_references). Call it DefaultL2MissRate.
(b) Find out the IPC (instruction per cycle). Call it BaselineIPC. (Here L1 latency = 1 cycle and L2 latency = 6 cycles).
(c) Now increase the latency of L1 cache (both IL1 and DL1) to 2 cycle and get IPC and call it slowL1IPC.
(d) Bring back latency of L1 cache to 1 cycle. Increase latency of L2 cache (which is data/instruction unified, so just change DL2 latency) to 7 cycle. Get IPC and call it slowL2IPC.
(e) Bring back L1 and L2 latency to their default values. Now change L2 replacement policy to random and FIFO and get L2 miss_rate with each of them. Call them RandomMissRate and FIFOMissRate, respectively.
(f) Bring back L2 replacement policy to LRU. Now, find IPC on changing issue policy to in-order, call it InOrderIPC.
(g) Bring back the issue policy to its original value. Change instruction decode, issue and commit bandwidth to 8, 8, 8, respectively. Find the new IPC, call it DiffWidthIPC

Submission format: Your submission should be just a RollNumber_Lab.txt file, e.g., 1111111_Lab1.txt. The format of this is provided below.

Student Name
Rollnumber
BenchmarkName
CacheSizeInKB
DefaultL2MissRate X.ABCD
BaselineIPC X.ABCD
slowL1IPC X.ABCD
slowL2IPC X.ABCD
RandomMissRate X.ABCD
FIFOMissRate X.ABCD
InOrderIPC X.ABCD
DiffWidthIPC X.ABCD

Note that if you cache size is 1MB, then CacheSizeInKB becomes:
1024

We will use automated scripts to check the answer, so do not change the format. Any change in the format will automatically make your answer wrong. No text or justification or comment should be provided in your submission. Do not approximate or round-off the answer: whatever answer you get from simulator should be reported exactly (4 digits after decimal).

Hints:
* This webpage http://faculty.cse.tamu.edu/djimenez/614-spring14/hw4/index.html provides some hints on downloading and compiling simplescalar.

* You need not modify the simulator code for this exercise: all the parameters can be specified as command-line arguments.

* You cannot do this lab-experiment on a mobile-phone.

* If you have linux, it will be easy to run this code. If you have windows, you can install windows subsystem for linux (https://docs.microsoft.com/en-us/windows/wsl/install-win10) or a virtualbox with Ubuntu, or a virtual machine

* Post your questions on teams