\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Requirements:**

Download the GEM5 configuration script for this problem along with cross-compiled executables for matrix multiplication and nqueens from (Exercises→Code→CH2). Copy the executables in your gem5/tests/test-progs directory. Here’s a brief description of the files

|  |  |
| --- | --- |
| File | Description |
| br\_pred.py | GEM5 Configuration script for simulating the workloads |
| matmul\_ijk\_64.out | Performs multiplication of two integer 64\*64 matrices |
| matmul\_ijk\_128.out | Performs multiplication of two integer 128\*128 matrices |
| nq8.out | Solves the n-queens problem for n=8 |
| nq10.out | Solves the n-queens problem for n=10 |

Make sure to edit the name of the executable in configs/tutorial/br\_pred.py before running the simulation

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**I.1:  Evaluating the performance of branch predictors available in GEM5**

For power ISA in the table below, make appropriate changes in configs/tutorial/caches.py to set L1D cache size, L2 cache size, and L2 data latency

|  |  |
| --- | --- |
|  | **Power 9** |
| **L1D cache size** | **32kB** |
| **L2 cache size** | **512kB** |
| **L2 data latency** | **20** |

When running the simulation, use the flag --bp-type=<predictor\_name> to specify the type of branch predictor. <predictor\_name> can be set to one of the following –(LocalBP, TournamentBP, TAGE)

For ex., the command below will simulate the workload with TAGE branch predictor and generate the simulation stats in out\_stats\_tage directory.

build/POWER/gem5.opt -d out\_stats\_tage configs/tutorial/br\_pred.py --bp-type=TAGE

Answer the following questions:

1. Simulate the provided .out files in GEM5 for three types of branch predictors – LocalBP, Tournament, and TAGE. Generate the stats.txt files for each simulation. Upload the stats.txt files on canvas. (You will have twelve (12) stats.txt files, three (3) for all three branch predictors for a single .out file. 12 in total for four .out files.
2. In your writeup, report the following metrics from stats.txt in a table for all 12 combinations of .out files and branch predicters,
3. system.cpu.branchPred.condPredicted
4. system.cpu.branchPred.condIncorrect

1. Is there a predictor which consistently performs better than others in terms of system.cpu.branchPred.condIncorrect ?
2. Based on the GEM5-based problems assigned in this assignment, write a detailed paragraph about extensions to this case study on the POWER memory hierarchy that could be pitched as potential (modest) project proposals.