Project Description

Championship Branch Prediction Infrastructure:
We will use the same simulation infrastructure that was used for the Branch Prediction Championship from ISCA 2011. The instructions for downloading and using the infrastructure are available here. The code should be compiled and run without changing the Makefile or any other existing files (except for predictor.h and predictor.cc). The code should NOT require any library code that is not part of C++. We will run the code using the shuttle cluster. If we cannot compile and run the code, we will not be able to evaluate your project.

Part 1: Implement Perceptron Branch Predictor (40 points)
You can read about the perceptron branch predictor here:
http://www.cs.utexas.edu/~lin/papers/hpca01.pdf
Use a budget of 8KB to implement a perceptron branch predictor where each weight counter is 8-bits and the threshold value for learning is zero. The perceptron entry is selected using a hash of the instruction address.
Vary the history length and number of entries in the table as follows:
1. 16-bit global history, 512-entry table
2. 32-bit global history, 256-entry table
3. 64-bit global history, 128-entry table

You must report the misprediction rate (mpki) for all the traces, as well as an arithmetic mean over all the traces.

Part 2: Branch Prediction Competition. (60 points, will be graded on basis of predictor performance)
Use a storage budget of 64KB (+256 bytes) to implement the branch predictor with lowest misprediction rate. You can used ideas that have been proposed in the past, some combination of those ideas, or develop your own ideas. You are however, specifically prohibited from using code from outside sources. All submitted code for the predictor must be written by you. The scoring for this part will be done on a competitive basis.
Submission Guide
1. For Part 1, Two files (**predictor.cc** and **predictor.h**) that implement the perceptron predictor. A 1-page report (**report.pdf**) showing the MPKI for perceptron predictor for different history length (as described above).
2. For Part 2, Two files (**predictor.cc** and **predictor.h**) that implement your entry to the competition. The report file (**report.pdf**) must be a conference-quality write-up of your proposed branch prediction algorithm, including references to relevant related work. The report must clearly describe how the algorithm works, and how it fits within the allocated storage budget. The report must be formatted as follows: no more than four pages, single-spaced, two-column format, minimum 10pt Times New Roman font. You must report the arithmetic mean of mpki over all traces.