To solve this assignment use the files in the following tarball:
/home/classes/ee4720/com/tca/hw3.tar.bz2. Instructions will be added to the procedures page on
how to do this on Wednesday morning.

Problem 1: Write a program which the local predictor will predict less accurately than the gshare
predictor. Verify that program using the pred analyzer. In the comments explain the idea behind
the program.

Problem 2: Design and test your own branch prediction algorithm by modifying the pred analyzer,
adding your new predictor to it. Test the predictor using the gcc and TeX benchmarks and a
benchmark of your own. Your benchmark, which you are encouraged to write yourself, should be
predicted more accurately using your predictor than any of the ones already simulated by pred.
Your predictor must be qualitatively different than the four already simulated. That is, you
cannot just tweak table sizes and the like.

Here are some ideas for new predictors.

- On a procedure return ghr may only contain callee history. Devise some way of restoring
  some caller history.

- While in a procedure branch behavior might correlate with the caller. Devise some way of
  including some caller information.

- Perhaps different ghr sizes (or none at all) could be used if the branch condition was computed
  a long time ago or was loaded from memory.

- The local history predictor is an inefficient way of predicting fixed iteration loops because the
table size is $2^{n-1}$, where $n$ is the most number of iterations that can be perfectly predicted.
Design a predictor that can handle much larger loops.