Gaurav Joshi
Parth Ray

Thumbsim Questions

1.If you are building a processor and have to do static branch prediction (meaning you have to assume at compile time whether a branch is taken or not), how should you do it? You can make a different decision for branches that go forward or backward.

If we were building a processor with static branch prediction, we would choose to predict taken for backwards branches and not taken for forward branches. This is due to the fact that when we ran matmul through our simulator, more forward branches were not taken. On the other, when we ran fib and chksum, there were more backwards branches taken than not taken.

2.If you are building a 256-byte direct-mapped cache, what should you choose as your block (line) size?

To find the block size that we would choose for building a 256-byte direct-mapped cache we ran each of the programs through the simulator and compared the hit rates of each block size. We determined that an 8 byte block size works the best for the bigger programs (With hit rates of 90.7341% for matmul and 96.9815% for chksum) and thus should be used for building the direct-mapped cache.

3.What conclusions can you draw about the differences between compiling with no optimization and -O2 optimization?

This question does not apply.