Roee Landesman and Eitan Simler

CPE 315: Computer Architecture

Thumb Simulator Writeup

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.
   a. Based on the data derived from the statistics gathered on the shang puzzle, I would choose to assume that the branch is taken. The branch taken occurs more often than the branch not taken in both forward branches (1.53 ratio in taken to not taken), and backward branches (21.715 ratio in taken to not taken).
2. If you are building a 256-byte direct-mapped cache, what should you choose as your block (line) size?
   a. The highest hit rate, as seen in the statistics gathered, is in the 32 byte block size for both Shang1 and Shang2 at approximately 85%. The highest hit rate was seen in the 8 byte block size for shang0, at 94%. Therefore, assuming that we are running a program with the highest optimization, we would choose the 32 byte block size.
3. What conclusions can you draw about the differences between compiling with no optimization and -O2 optimization?
   a. The number of dynamic instructions was decreased by ~80% from Shang0 to Shang2, demonstrating the compiler's ability to optimize CPI.
   b. The statistics count decreased for Shang2, with the decreased number of dynamic instructions being a huge factor.
   c. The hit rate for shang0 are generally higher than that of shang2, however on average, due to the lower CPI, the program was still faster in completion time.
   d. The forward branches not taken in shang2 are relatively high in comparison to that of the forward branches not taken in shang0.

Github (Roee): github.com/rlandesman/ComputerArchitecture
Github (Eitan): github.com/Lupo-CPE315-Spring18/thumbsim-the-vegetarian-burrito-fan-club