**ZACHARY HALPERN**

**CS320 – Fall 2017**

**Homework Assignment 1**

**Due: Friday, September 8th in class**

1. Consider program P, which runs on a 4 GHz machine M in 500 seconds. Consider an optimization to P that replaces all instances of multiplying a value by 4 (*mult X,X,4*) with two instructions that set x to x+x twice (*add X,X; add X,X*). Assume that every multiply instruction takes 4 cycles to execute, and every add instruction takes just 1 cycle. After recompiling, the program now runs in 400 seconds on machine M. Determine how many multiplies were replaced by this optimization. In your answer, show all of your calculations and analysis. [20%]

**Cycles in P: 500sec \* 4x109 cycles/sec = 2x1012 cycles**

**Cycles in P’: 400sec \* 4x109 cycles/sec = 1.6x1012 cycles**

**Cycle difference: 2x1012 – 1.6x1012 = 0.4x1012**

**We had X multiplication instructions which took 4X time and now 2X addition instructions that take 2X time.**

**2X = 0.4x1012 => X = 0.2x1012 = 2x1011 multiplication instructions.**

1. Your company could speed up a Java program on their new computer by adding hardware support for garbage collection. Assume that garbage collection currently comprises 50% of the cycles of the program. You have two possible changes to consider. The first one would be to automatically handle garbage collection in hardware, causing the increase in cycle time by a factor of 1.2. The second option would be to add the new instructions to the ISA that could be used during garbage collection. This would halve the number of cycles needed for garbage collections, but would increase the cycle time by a factor of 1.1. Which of these two options, if any, should you choose? [20%]

**TimeOLD = Clock Cycles \* Clock Cycle Time.**

**TimeOPTION1 = (1 - 0.5) \* Clock Cycle Time \* 1.2 = 0.6\*Clock Cycle Time**

**TimeOPTION2 = (1 - (0.5 / 2)) \* Clock Cycle Time \* 1.1 = 0.825\*Clock Cycle Time**

**Better option is to use the hardware route, as it will only take 60% of the time to run compared to now.**

1. Consider two possible improvements that can be used to enhance a machine. You can either make multiply instructions run four times faster than before, or make memory access instructions run two times faster than before. You repeatedly run a program that takes 100 seconds to execute. Of this time, 25% is used for multiplication, 40% for memory access instructions, and 35% for other tasks.

* 1. What will the speedup be if you improve only multiplication? [5%]

**Timproved = 25sec/4 + 75sec = 81.25sec.**

**100/81.25 = 1.23x speedup.**

* 1. What will the speedup be if you improve only memory access? [5%]

**Timproved = 40sec/2 + 60sec = 80sec.**

**100/80 = 1.25x speedup.**

* 1. What will the speedup be if both improvements are made? [10%]

**Timproved = 25sec/4 + 40sec/2 + 35sec = 61.25sec.**

**100/61.25 = 1.63x speedup.**

1. Assume that multiply instructions take 4 cycles to execute and account for 20% of the instructions in a typical program and that the other 80% of the instructions require an average of 2 cycles for each instruction.
   1. What is the percentage of time that the CPU spends doing multiplication? [5%]

**TimeMULT = 0.2\*IC\*4 cycles per instruction\*Cycle Time**

**= 0.8\*IC\*Cycle Time**

**TimeALL = (0.2\*IC\*4 cycles per instruction + 0.8\*IC\*2 cycles per instruction)\*Cycle Time**

**= (0.8\*IC + 1.6\*IC)\*Cycle Time**

**= 2.4\*IC\*Cycle Time**

**Percent increase = 0.8/2.4 = 0.33 = 33.3% of the time doing Multiplication.**

* 1. Estimate the CPI value for a typical program executing on this machine [5%].

**80% take 2 cycles, 20% take 4 cycles. So 0.8\*2 + 0.2\*4 = 2.4 cycles per instruction avg.**

* 1. Assume that it is possible to reduce the number of cycles required for multiplication from 4 to 2, but this will require a 20% increase in the cycle time. Nothing else will be affected. Should we proceed with this modification? Explain why or why not [10%].

**This modification should not be proceeded with since it will have no impact on the overall time.**

**TimeOLD = 2.4\*IC\*Cycle Time**

**TimeNEW = (0.2\*IC\*2Cycles Per Instruction + 0.8\*IC\*2CPI)\*Cycle Time\*1.2**

**TimeNEW = 2\*IC\*Cycle Time\*1.2 = 2.4\*IC\*Cycle Time**

**As we can see, the results from TimeOLD and TimeNEW are equal.**

1. Suppose that we designed a new floating-point unit that accelerates the floating-point instructions by a factor of 5. We are now looking for a benchmark to show off the new floating-point unit and we want the overall benchmark to show a speedup of 3. One benchmark we are considering runs for 100 seconds with the old floating-point hardware. How much of the execution time would floating-point instructions have to account for in this program in order to yield our desired speedup on this benchmark? [20%]

**FTIME = execution time of floating point instructions**

**Timproved = Taffected / factor + Tunaffected**

**100sec / 3factor = FTIME / 5factor + (100 – FTIME)**

**FTIME = 83.3 sec.**