Aiden Cullo

446 Circle Avenue, Kingston, NY 12401 — 845.943.8311 — culloaiden3@gmail.com — http://www.aidencullo.me

1. **Question:** 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%]

   Answer: We can use Amdahl's Law to calculate the number of multiple instructions have been substituted with two add instructions.
   The improved time is the resulting time:

   $$T_{\text{Improved}} = 400\, seconds$$

   The Affected time is the time subject to improvement which is complmentary to unaffected time

   $$T_{\text{Affected}} = 500 - x\, seconds$$

   The Unaffected time cannot be shortened

   $$T_{\text{Unaffected}} = x\, seconds$$

   The improvement factor is the ratio of the changed instruction count to the previous instruction count

   $$n = 2$$
   $$T_{\text{Improved}} = \frac{T_{\text{Affected}}}{2} + T_{\text{Unaffected}}$$
   $$400 = \frac{500 - x}{2} + x$$
   $$800 = 500 - x + 2x$$
   $$x = 300$$
   $$T_{\text{Affected}} = 200\, seconds$$

   If the affected time is 200 seconds on a 4GHz machine, it will run 8,000,000,000 (trillion) cycles and therefore 2,000,000,000 (trillion) multiple instructions need to be converted to add instructions

2. **Question:** 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%]

   Answer: If $CPU\, Time\, (CPU) = Instruction\, Count\, (IC) \times Cycles\, Per\, Instruction\, (CPI) \times Seconds\, Per\, Cycle\, (SPC)$, we can calculate the effect of each option on the total CPU time.

   $$CPU_{\text{first}} = IC_{\text{first}} \times CPI_{\text{first}} \times SPC_{\text{first}}$$
   $$CPU_{\text{first}} = IC \times CPI \times (SPC \times 1.2)$$

   $$CPU_{\text{second}} = IC_{\text{second}} \times CPI_{\text{second}} \times SPC_{\text{second}}$$

$$CPU_{\text{second}} = \frac{IC}{2} \times CPI \times (SPC \times 1.1)$$

$$CPU_{\text{second}} \,?\, CPU_{\text{first}}$$
$$.55 \times CPU \,?\, 1.2 \times CPU$$
$$.55 \times CPU < 1.2 \times CPU$$

Therefore, there second option will reduces the CPU time by a factor of 1.82 while the first option will increase the CPU by a factor of 1.2

3. **Question:** 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.

   (a) What will the speedup be if you improve only multiplication? [5%]
   Answer: If the program runs for 100 seconds, then it runs multiplication instructions for 25 seconds, memory access for 40 seconds and other instructions for 35 seconds. We can use the same equation from question one by setting the spead-up time to be 25 seconds and the unaffected time 75. seconds.

$$T_{\text{multiply}} = \frac{25\,seconds}{4} + 75\,seconds$$
$$T_{\text{multiply}} = 81.25\,seconds$$

$$Speedup = 100/81.25$$

   (b) What will the speedup be if you improve only memory access? [5%]
   Answer: For just memory access the speedup time will be 40 seconds over an improvement factor of two with 60 seconds unaffected.

$$T_{\text{memory}} = \frac{40\,seconds}{2} + 60\,seconds$$
$$T_{\text{memory}} = 80\,seconds$$

$$Speedup = 100/80$$

   (c) What will the speedup be if you both improvements are made? [5%]
   Answer: For both improvments we set the spead-up time to be 25 seconds over a improvement factor of 4 and 40 seconds over an improvement factor of 2 with the unaffected time as 35 seconds.

$$T_{\text{total}} = \frac{25\,seconds}{4} + \frac{40\,seconds}{2} + 35\,seconds$$
$$T_{\text{total}} = 61.25\,seconds$$

$$Speedup = 100/61.25$$

4. **Question:** 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.

   (a) What is the percentage of time that the CPU spends doing multiplication? [5%]
   Answer: If 20% of the instructions are multiplication there will be $20 \times 4 = 80$ cycles for of from

   multiplication instructions and $80 \times 2 = 160$ cycles from other intructions. Therefore, $\frac{80}{80+160} = 33.3\%$ of the time was spent on multiply instructions.

   (b) Estimate the CPI value for a typical program executing on this machine [5%].
   Answer: $CPI = \frac{240\,(total\,number\,of\,instructions)}{100\,(total\,number\,of\,seconds)} = 2.4\,cycles\,per\,second$

   (c) 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%].
   Answer: No, since decreasing the number of cycles for multiplication will only reduce the total CPU time by 1.11 (multiply it by .9) because multiply is only 20% of the instructions so 80% will be unaffected.

$$T_{\text{Improved}} = \frac{20}{2} + 80 = 90$$
$$90/100 = .9$$

   With an increase in cycle time this modification will be unfavorable because the total CPU time will increase

$$CPU_{\text{after}} = IC \times (CPI \times 0.9) \times (SPC \times 1.2)$$
$$CPU_{\text{after}} = (CPU_{\text{before}} \times 1.08)$$

5. **Question:** 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%].
   Answer: From a speedup of 3 we will get an improved time of 100/3 seconds. We plug into Amdahl's Law

$$T_{\text{Improved}} = \frac{T_{\text{Affected}}}{5} + T_{\text{Unaffected}}$$
$$\frac{100}{3} = \frac{100 - x}{5} + x$$
$$x = \frac{200}{12} = 16.7$$

   Therefore, 100-16.7 = 83.3 seconds of the program has to be floating points instructions to get the desired speedup.