

Question 1: (0 points)
Bank of Questions

Performance Analysis Because of Changes (V0F, V10, V11)

You have been hired by a very important supplier of computer servers for Internet Service Providers (ISPs). The most popular program run by ISPs has a CPI of 1.5 clocks per instructions. This high CPI is dominated by a class of instructions C_i . Instructions belonging to C_i are executed 40% of the time and, on average, each instruction of class C_i takes 3 cycles to execute. Lets call the execution time of this original machine T_{orig} .

A processor-architecture team in your company has proposed a design change to the processor that will reduce the average number of clocks executed by C_i instructions to 2 clock cycles, but will increase the clock cycle by 20%. Lets call the execution time of this machine T_{proc}

The head of your compiler team proposed that, with a larger development team, they could implement compiler optimizations that would reduce the number of C_i instructions executed by 35% without affecting the number of execution of instructions of other classes. Lets call the execution time of this machine T_{comp}

Given budget constraints, you only can afford to approve one of these changes.

Question 2: (8 points)
Which solution is faster and by how much?

Question 3: (8 points)
How much faster is the better solution when compared with the original machine?

Solution: Let:

I_X : number of instructions executed in machine X

T_X : execution time in machine X

C_X : Clock cycle in machine X

CPI_X : CPI in machine X

$CPI_{X:others}$: CPI for other instructions in machine X

$X = orig$ for original machine;

$X = proc$ for the machine where the processor is modified and

$X = axes = comp$ for the machine where the compiler is modified.

$$T_X = I_X \times CPI_X \times C_X \quad (1)$$

$$T_{orig} = I_{orig} \times CPI_{orig} \times C_{orig} \quad (2)$$

$$= I_{orig} \times 1.5 \times C_{orig} \quad (3)$$

$$CPI_{orig} = 0.6 \times CPI_{orig:others} + 0.4 \times 3 = 1.5 \frac{\text{clocks}}{\text{instruction}} \quad (4)$$

$$CPI_{orig:others} = \frac{1.5 - 0.4 \times 3}{0.6} = 0.5 \frac{\text{clocks}}{\text{instruction}} \quad (5)$$

$$CPI_{proc} = 0.6 \times CPI_{orig:others} + 0.4 \times 2 = 0.6 \times 0.5 + 0.4 \times 2 = 1.1 \frac{\text{clocks}}{\text{instruction}} \quad (6)$$

$$T_{proc} = I_{proc} \times CPI_{proc} \times C_{proc} \quad (7)$$

$$T_{proc} = I_{orig} \times 1.1 \times 1.2 \times C_{orig} \quad (8)$$

$$= 1.32 \times I_{orig} \times C_{orig} \quad (9)$$

Assume that 1000 instructions were executed in the original machine. Therefore there were 400 instructions of class C_i and 600 instructions from other classes. The compiler reduced the number of instructions of class C_i by 35%, therefore now there are 260 instructions of the class C_i . We computed above the CPI of others to be $0.5 \frac{\text{clocks}}{\text{instruction}}$ and that value does not change. Therefore:

$$CPI_{comp} = \frac{600 \times 0.5 + 260 \times 3}{600 + 260} = 1.26 \frac{\text{clocks}}{\text{instruction}} \quad (10)$$

$$T_{comp} = (0.6 + 0.4 \times 0.65) \times I_{orig} \times CPI_{comp} \times C_{orig} \quad (11)$$

$$T_{comp} = 0.860 \times I_{orig} \times 1.26 \times C_{orig} \quad (12)$$

$$T_{comp} = 1.08 \times I_{orig} \times C_{orig} \quad (13)$$

Therefore,

A. The compiler solution is $\frac{1.32}{1.08} = 1.22$ times faster than the processor-architecture solution.

B. The compiler solution is $\frac{1.5}{1.08} = 1.38$ times faster than the original machine.