Consider three different processors P1, P2, and P3 executing the same instruction set. P1 has a 3 GHz clock rate and a CPI of 1.5. P2 has a 2.5 GHz clock rate and a CPI of 1.0. P3 has a 4.0 GHz clock rate and has a CPI of 2.2.
a. Which processor has the highest performance expressed in instructions per second?
IPS = Clock Rate / CPI
P1: 3/1.5 = 2
P2: 2.5/1 = 2.5
P3 = 4/2.2 = 1.81 repeating
P2 has the highest performance.

b. If the processors each execute a program in 10 seconds, find the number of cycles and the number of instructions.
Cycles(Clock rate * time):
P1: 3 x 10^9 (GHz) * 10 seconds = 10x10^10 seconds
P2: 2.5 x 10^9 (GHz) * 10 seconds = 2.5x10^10 seconds
P3: 4 x 10^9 (GHz) * 10 seconds = 4 x10^10 seconds
Instructions (Cycles / CPI):
P1: P1C/1.5 = 2 x 10^10
P2: P2C/1 = 2.5 x 10^10
P3: P3C/2.2 = 1.81 x 10^10
c. We are trying to reduce the execution time by 30% but this leads to an increase of 20% in the CPI. What clock rate should we have to get this time reduction?
We can only change the clock rate:
Time = (Instructions * CPI) / Clock Rate
Instructions is constant, cpi = cpi*1.2, time = .7time
(.7 * cpi)/ clockrate = (1.2*cpi)/ new clock rate
.7/clockrate = 1.2/new clock rate
New clock rate = 1.714clockrate
~171.4% = 71.4% increase to the clock rate.

Assume a program requires the execution of 50 x 10^6 FP instructions, 110 x 10^6 INT instructions, 80 x 10^6 L/S instructions, and 16 x 10^6 branch instructions. The CPI for each type of instruction is 1, 1, 4, and 2, respectively. Assume that the processor has a 2 GHz clock rate.
(a) By how much must we improve the CPI of FP instructions if we want the program to run two times faster?
Current execution time: ((50*1 + 110*1 + 80*4 + 16*2)*10^6) / (2*10^9) = .256 seconds. To cut that in half we need to do it in .128 seconds. If FP is the only thing that can be changed, we can add up the cycles that aren't FP (everything after 50*1) to find the new CPI: 50x + 462 = 256. Even before solving this I can

obviously see it's impossible, unless we somehow have a negative speed, so I don't think it's possible if we're only changing the FP CPI. Even if it's 0 it would still be 462, it would have to be -4.12.

(b) By how much must we improve the CPI of L/S instructions if we want the program to run two times faster?

This one is actually doable, this time we just skip out on the 80: 192 + 80x = 256, we find that x = .8 which means the new CPI of the L/S needs to be .8.

(c) By how much is the execution time of the program improved if the CPI of INT and FP instructions is reduced by 40% and the CPI of L/S and Branch is reduced by 30%?

FP = 1 * .6 = .6
INT = 1 * .6 = .6
L/S = 4 * .7 = 2.8
Branch = 2 * .7 = 1.4

50(.6) + 110(.6) + 80(2.8) + 16(1.4) = 342.4 million cycles:
(342.4 x 10^6) / (2 x 10^9) = .1712 seconds

Since the original time was .256:
.256/.1712 = 1.495 times faster.