Task 1: Quantitative Problem Solving (E1)

Problem:
After graduating, you are asked to become the lead computer designer at Hyper Computers Inc. Your study of usage of high-level language constructs suggests that procedure calls are one of the most expensive operations. You have invented a scheme that reduces the loads and stores normally associated with procedure calls and returns. The first thing you do is run some experiments with and without this optimization. Your experiments use the same state-of-the-art optimizing compiler that will be used with either version of the computer. These experiments reveal the following information:

- The clock rate of the unoptimized version is 5% higher.
- 30% of the instructions in the unoptimized version are loads or stores.
- The optimized version executes 2/3 as many loads and stores as the unoptimized version. For all other instructions, the dynamic counts are unchanged.
- All instructions (including load and store) take one clock cycle.

Which is faster? Justify your decision quantitatively.

Solution:

To determine which version of the computer is faster, we need to compare the execution times of the optimized and unoptimized versions. We can use the CPU performance equation:

$$\text{CPU Time} = \text{Instruction Count} \times \text{CPI} \times \text{Clock Cycle Time}$$

Given:

- The clock rate of the unoptimized version is 5% higher than the optimized version.
- 30% of the instructions in the unoptimized version are loads or stores.
- The optimized version executes 2/3 as many loads and stores as the unoptimized version.
- All instructions take one clock cycle.

Let's denote:

- $f_{\text{load/store}}$ as the fraction of load/store instructions.
- f_{other} as the fraction of other instructions.
- $\text{Clock Rate}_{\text{unopt}}$ as the clock rate of the unoptimized version.
- $\text{Clock Rate}_{\text{opt}}$ as the clock rate of the optimized version.
- $\text{Instruction Count}_{\text{unopt}}$ as the instruction count of the unoptimized version.
- $\text{Instruction Count}_{\text{opt}}$ as the instruction count of the optimized version.

From the problem:

- $f_{\text{load/store}} = 0.30$
- $f_{\text{other}} = 1 - f_{\text{load/store}} = 0.70$

- $\text{Clock Rate}_{\text{opt}} = \text{Clock Rate}_{\text{unopt}} \times 0.95$
- The optimized version executes $\frac{2}{3}$ as many load/store instructions as the unoptimized version.

The instruction count for the unoptimized version is:

$$\text{Instruction Count}_{\text{unopt}} = f_{\text{load/store}} \times \text{Total Instructions} + f_{\text{other}} \times \text{Total Instructions}$$

The instruction count for the optimized version is:

$$\text{Instruction Count}_{\text{opt}} = (\frac{2}{3} \times f_{\text{load/store}}) \times \text{Total Instructions} + f_{\text{other}} \times \text{Total Instructions}$$

Let's denote the total number of instructions as I:

$$\text{Instruction Count}_{\text{unopt}} = 0.30I + 0.70I = I$$

$$\text{Instruction Count}_{\text{opt}} = (\frac{2}{3} \times 0.30I) + 0.70I = 0.20I + 0.70I = 0.90I$$

The clock cycle time for the optimized version is:

$$\text{Clock Cycle Time}_{\text{opt}} = \frac{1}{\text{Clock Rate}_{\text{opt}}} = \frac{1}{0.95 \times \text{Clock Rate}_{\text{unopt}}}$$

The clock cycle time for the unoptimized version is:

$$\text{Clock Cycle Time}_{\text{unopt}} = \frac{1}{\text{Clock Rate}_{\text{unopt}}}$$

Now, we can calculate the CPU time for both versions:

$$\text{CPU Time}_{\text{unopt}} = I \times 1 \times \text{Clock Cycle Time}_{\text{unopt}} = I \times \frac{1}{\text{Clock Rate}_{\text{unopt}}}$$

$$\text{CPU Time}_{\text{opt}} = 0.90I \times 1 \times \text{Clock Cycle Time}_{\text{opt}} = 0.90I \times \frac{1}{0.95 \times \text{Clock Rate}_{\text{unopt}}}$$

Simplifying the CPU time for the optimized version:

$$\text{CPU Time}_{\text{opt}} = 0.90I \times \frac{1}{0.95} \times \frac{1}{\text{Clock Rate}_{\text{unopt}}} = \frac{0.90I}{0.95} \times \frac{1}{\text{Clock Rate}_{\text{unopt}}}$$

Comparing the CPU times:

$$\text{CPU Time}_{\text{opt}} = \frac{0.90I}{0.95} \times \frac{1}{\text{Clock Rate}_{\text{unopt}}} \approx 0.947I \times \frac{1}{\text{Clock Rate}_{\text{unopt}}}$$

$$\text{CPU Time}_{\text{unopt}} = I \times \frac{1}{\text{Clock Rate}_{\text{unopt}}}$$

The improvement in CPU time for the optimized version is:

$$\text{Improvement} = (1 - \frac{\text{CPU Time}_{\text{opt}}}{\text{CPU Time}_{\text{unopt}}}) \times 100\%$$

$$\text{Improvement} = (1 - \frac{0.947I}{I}) \times 100\%$$

$$\text{Improvement} = (1 - 0.947) \times 100\%$$

$$\text{Improvement} = 0.053 \times 100\%$$

$$\text{Improvement} = 5.3\%$$

Conclusion:

The optimized version of the computer is faster by approximately 5.3%.