COMP4300 Computer Architecture

Name: Mohab Yousef.

Fall 2023.

Homework 1

Due Monday Sept 25, 11:59pm

1. (15 points) Suppose that a given optimization results in an OVERALL

speedup of 1.5 over the original design. If the optimization speeds up loads

and stores, which collectively accounted for 75% of the execution time

BEFORE the optimization, by what factor were loads and stores sped up by

the optimization?

Speed up over all = f = 0.75, speedup =1.5

1.5 = , 1.5 (0.25 + 0.75/s) = 1.

0.375 + 1.125/s = 1, 1.125/s = 0.625.

0.625s = 1.125.

S for loads and stores = 1.8

2. (15 points) If, in problem one, the described optimization could make loads

and stores take no time at all (not realistic, just for the sake of argument),

what would the overall speedup to the execution time be?

In this case f/s =

= = = -1.33.

3. (30 points) Suppose that in problem one, the fraction of time taken for loads

and stores AFTER optimization is 25%, and overall speedup is 1.5, what

would the fraction of time taken by loads and stores BEFORE optimization?

What would the speedup to loads and stores be?

Speed up over all = f = 0.75, speedup =1.5

1.5 = , 1.5 (0.75 + 0.75/s) = 1.

1.125 + 1.125/s = 1, 1.125/s = 1.125.

S=1

Speedup overall = Execution time old/ Execution time new

Speed up to load = 0.75/0.25 = 3

Free/s = 0.25 what we looking for is free so instead of (0.75/s) = 0.25 and we find the 0.25

4. (30 points) For a PDP-8, generate assembly code and binary to multiply the

number in hex address 0x200 by 4, and store the result in address 0x201.

The program should start in address 0x100. You can assume the number in

0x200 is positive and less than 0x100. You will need to consult the Internet

for the PDP-8 mnemonics and instruction formats. Note in particular that the

PDP-8 has no multiply instruction. Be sure to give the address of each

instruction.

|  |  |  |
| --- | --- | --- |
| Address in hex | mnemonic | binary |
|  |  |  |
| 0x100 | CLA | 111 010 000 000 |
| 0x101 | TAD  0x105 | 001 110 000 101 |
| 0x102 | RTL | 111 000 000 110 |
| 0x103 | DCA 0x106 | 011 110 000 110 |
| 0x104 | HLT | 111 100 000 010 |
| 0x105 | Data stored 0x200 | 001 000 000 000 |
| 0x106 | DCA 0x201 | 001 000 000 001 |

5. (10 points) If a given processor has a dynamic instruction count that is 20%

ALU instructions, 40% loads and stores, and 20% jumps, and if ALU

Instructions take 1 cycle to execute, load or store take 5 cycles, jumps take 3

cycles, and on average all the other instructions not listed take 2.3 cycles,

what is the average CPI for the processor?

Average CPI = (200 \* 1 + 400 \* 5 + 200 \* 3 + 200 \* 2.3) / 1000

Average CPI = (200 + 2000 + 600 + 460) / 1000

Average CPI = 3260 / 1000

Average CPI = 3.26