CSE 321a: Computer Organization (1)
Third Year, Computer & Systems Engineering

Assignment #1

Due date: **Wednesday, 19th October, 2016**

1. Consider two different processors, P_1 and P_2, that implement …

(a) How many cycles each processor takes on average to execute one instruction from program X?

Time taken by P_1 to execute program X = $T_1 = N_X * CPI_1 / f_1$
Average CPI in program X (by P_1) = $CPI_1 = T_1 * f_1 / N_X = (30 * 10^{-3}) * (200 * 10^6) / 10^6 = 6$
Time taken by P_2 to execute program X = $T_2 = N_X * CPI_2 / f_2$
Average CPI in program X (by P_2) = $CPI_2 = T_2 * f_2 / N_X = (31.25 * 10^{-3}) * (160 * 10^6) / 10^6 = 5$

(b) What is the number of each class of instructions in program X?

Average CPI in program X (by P_1) = $CPI_1 = Ratio_A * CPI_{A1} + Ratio_B * CPI_{B1} + Ratio_C * CPI_{C1}$
$6 = Ratio_A * 4 + Ratio_B * 9 + Ratio_C * 6$ ==> equation (1)
Average CPI in program X (by P_2) = $CPI_2 = Ratio_A * CPI_{A2} + Ratio_B * CPI_{B2} + Ratio_C * CPI_{C2}$
$5 = Ratio_A * 5 + Ratio_B * 10 + Ratio_C * 3$ ==> equation (2)
Since all ratios must add up to 1:
$1 = Ratio_A + Ratio_B + Ratio_C$ ==> equation (3)
Solving equations (1), (2), and (3):
$Ratio_A = 0.3$, $Ratio_B = 0.2$, and $Ratio_C = 0.5$
Number of class A instructions = $N_A = Ratio_A * N_X = 0.3 * 10^6 = 300000$
Number of class B instructions = $N_B = Ratio_B * N_X = 0.2 * 10^6 = 200000$
Number of class C instructions = $N_C = Ratio_C * N_X = 0.5 * 10^6 = 500000$

2. In a hypothetical computer, the processor has four registers …

(a) **Which memory locations can be used to store instructions, and which memory locations can be used to store data?**

PC is 16-bit long ➔ Instructions can be stored in memory locations 0000 to FFFF.
Address field in instructions is 14-bit long ➔ Data can be stored in memory locations 0000 to 3FFF.

(b) Show the execution trace of that program.

Instruc-tion	Cycle	PC	AC	CTR	PTR	Location: 3C98	Location: 3C99	Location: 3C9A	Location: 3C9B
Initially	---	D47C	C0FFE	3E0D	02B0	8000C	00003	80004	0BEEF
04000	Fetch	D47D	C0FFE	3E0D	02B0	8000C	00003	80004	0BEEF
	Execute	D47D	00000	3E0D	02B0	8000C	00003	80004	0BEEF
0FC98	Fetch	D47E	00000	3E0D	02B0	8000C	00003	80004	0BEEF
	Execute	D47E	00000	3E0D	3C98	8000C	00003	80004	0BEEF
14003	Fetch	D47F	00000	3E0D	3C98	8000C	00003	80004	0BEEF
	Execute	D47F	00000	0003	3C98	8000C	00003	80004	0BEEF

33FFF*	Fetch	D480	00000	0003	3C98	8000C	00003	80004	0BEEF
	Execute	D480	00090	0003	3C98	8000C	00003	80004	0BEEF
C8001	Fetch	D481	00090	0003	3C98	8000C	00003	80004	0BEEF
	Execute	D481	00090	0003	3C99	8000C	00003	80004	0BEEF
D7FFE	Fetch	D482	00090	0003	3C99	8000C	00003	80004	0BEEF
	Execute	D482	00090	0002	3C99	8000C	00003	80004	0BEEF
D8004	Fetch	D483	00090	0002	3C99	8000C	00003	80004	0BEEF
	Execute	D47F	00090	0002	3C99	8000C	00003	80004	0BEEF
33FFF*	Fetch	D480	00090	0002	3C99	8000C	00003	80004	0BEEF
	Execute	D480	00099	0002	3C99	8000C	00003	80004	0BEEF
C8001	Fetch	D481	00099	0002	3C99	8000C	00003	80004	0BEEF
	Execute	D481	00099	0002	3C9A	8000C	00003	80004	0BEEF
D7FFE	Fetch	D482	00099	0002	3C9A	8000C	00003	80004	0BEEF
	Execute	D482	00099	0001	3C9A	8000C	00003	80004	0BEEF
D8004	Fetch	D483	00099	0001	3C9A	8000C	00003	80004	0BEEF
	Execute	D47F	00099	0001	3C9A	8000C	00003	80004	0BEEF
33FFF*	Fetch	D480	00099	0001	3C9A	8000C	00003	80004	0BEEF
	Execute	D480	000A9	0001	3C9A	8000C	00003	80004	0BEEF
C8001	Fetch	D481	000A9	0001	3C9A	8000C	00003	80004	0BEEF
	Execute	D481	000A9	0001	3C9B	8000C	00003	80004	0BEEF
D7FFE	Fetch	D482	000A9	0001	3C9B	8000C	00003	80004	0BEEF
	Execute	D482	000A9	0000	3C9B	8000C	00003	80004	0BEEF
D8004	Fetch	D483	000A9	0000	3C9B	8000C	00003	80004	0BEEF
	Execute	D483	000A9	0000	3C9B	8000C	00003	80004	0BEEF
38000	Fetch	D484	000A9	0000	3C9B	8000C	00003	80004	0BEEF
	Execute	D484	0000D	0000	3C9B	8000C	00003	80004	0BEEF
27C9B*	Fetch	D485	0000D	0000	3C9B	8000C	00003	80004	0BEEF
	Execute	D485	0000D	0000	3C9B	8000C	00003	80004	0000D

(c) What does the program compute?

- This program calculates the **length of a vector** (whose elements are stored in memory from location 3C98 to location 3C9A) and saves the result to location 3C9B, or
- This program calculates the **distance between the origin and a point in space** (whose coordinates are stored in memory from location 3C98 to location 3C9A) and saves the result to location 3C9B.

(d) How much time does it take for the program to be executed? Justify your answer.

This hypothetical machine has only two types of instructions:

1. Instructions that don't have memory operands (e.g., C8001 and 38000). Each of these instructions takes 8 cycles: 5 (instruction fetch) + 3 (instruction processing).
2. Instructions that have one memory operand (e.g., 33FFF and 27C9B). Each of these instructions (marked by * in the table) takes 13 cycles: 5 (instruction fetch) + 3 (instruction processing) + 5 (operand fetch or store).

Total number of cycles = (13 instructions) * (8 cycles) + (4 instructions) * (13 cycles) = 156 cycles
Clock period = 1 / (100 MHz) = 10 nS
Total time = (156 cycles) * (10 nS) = 1.56 µS