NYCU Computer Organization HW 6

1. Consider the following code segment in C:

   a = b + c;

   c = b + f;

   Here is the generated RISC-V code for this segment, assuming all variables are in memory and are addressable as offsets from x31:

`lw`	`x1, 0(x31)`	`// load b`
`lw`	`x2, 8(x31)`	`// load e`
`add`	`x3, x1, x2`	`// b + e`
`sw`	`x3, 24(x31)`	`// store a`
`lw`	`x4, 16(x31)`	`// load f`
`add`	`x5, x1, x4`	`// b + f`
`sw`	`x5, 32(x31)`	`// store c`

   Please find the hazards in the preceding code segment and reorder the instruction to avoid any pipeline stalls.

2. Assume that the logic blocks used to implement a processor's datapath have the following latencies:

IMEM/D MEM	Register file	MUX	ALU	Adder	Single Gate	Register Read	Register Setup	Sign extend	Control
250 ps	150 ps	25 ps	200 ps	150 ps	5 ps	30 ps	20 ps	50 ps	50 ps

   "Register Read" is the time needed after the rising clock edge for the new register value to appear on the output. This value applies to the PC only. "Register Setup" is the amount of time a register's data input must be stable before the rising edge of the clock. This value applies to both the PC and Register File.

   2.1 What is the latency of an R-Type instruction (ps) (i.e. how long must the clock period be to ensure that this instruction works correctly?)

   2.2 What is the latency of `lw` ?

   2.3 What is the latency of `sw`?

   2.4 What is the latcny of `beq`?

3. Consider the following instruction mix

R-Type	I-Type (non-lw)	Load	Store	Branch	Jump
24%	28%	25%	10%	11%	2%

   3.1 What fraction of all instructions use data memory?

   3.2 What fraction of all instructions use instruction memory?

   3.3 What fraction of all instructions use the sign extend?