# Discussion 10: Pipelining

1. Consider the following set of instructions:

```
add $sp, $sp, -4
sub $v0, $a0, $a1
lw  $t0, 4($sp)
or  $s0, $s1, $s2
lw  $t1, 8($sp)
```

Assuming the instructions are executed on a **single-cycle machine** with 10ns cycle time, compute:

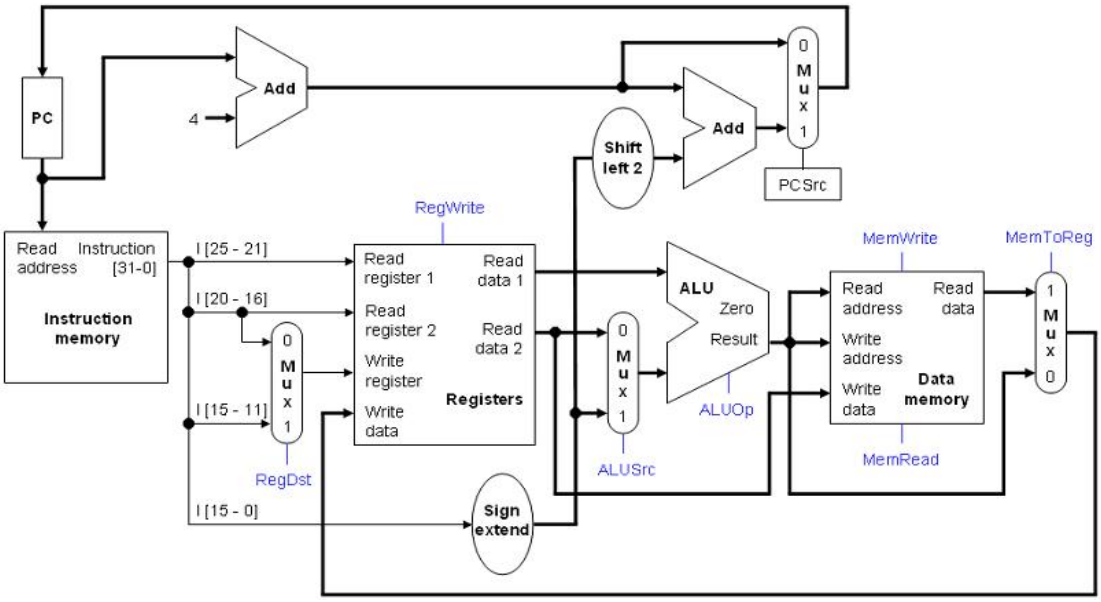
(a) cycle time (hint: this is *not* a trick question)

(b) instruction latency

(c) instruction throughput

2. Assume that the code above is executed on a 5-stage pipelined machine discussed in lecture. You might first draw the pipeline diagram in the space below.

If a pipeline stage takes 2ns, calculate:
(1) cycle time
(2) instruction latency
(3) instruction throughput




Suppose the functional units in the above datapath have the following latencies: Memory 5 ns, ALU 5 ns, Register file 3 ns. All other units have negligible latencies. Consider a sequence of $N$ instructions with no data or control dependencies.

**Problem 1**: How long does it take to execute these instructions on a single-cycle MIPS implementation? If the ALU is made faster by a factor of 1.25, how much would this improve performance?

**Problem 2**: How long does it take to execute these instructions on the 5-stage pipeline IF ID EX MEM WB described in class? If the ALU is made faster by a factor of 1.25, how much would this improve performance?

**Problem 3**: Suppose the IF and ID stages were merged to form a 4-stage pipeline. How long would the instructions take now? If the ALU is made faster by a factor of 1.25, how much would this improve performance?

**Problem 4**: Complete the pipeline diagram for the following sequence of instructions and identify *all* data dependencies on the 5-stage MIPS pipeline. Show that *all* data dependencies can be eliminated by carefully rearranging the code and renaming registers. With this optimization, how many cycles are needed to complete these instructions on the 5-stage pipeline?

```
                                  Cycle
                    1   2   3   4   5   6   7   8   9  10  11  12  13  14
                   ---------------------------------------------------------
li   $v0, 0         IF  ID  EX MEM  WB
lw   $t0, 0($a0)
add  $v0, $v0, $t0
lw   $t0, 4($a0)
add  $v0, $v0, $t0
lw   $t0, 8($a0)
add  $v0, $v0, $t0
lw   $t0, 12($a0)
add  $v0, $v0, $t0
```