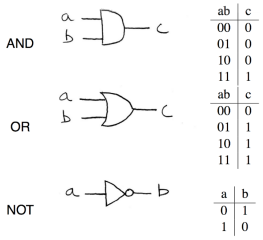
Student ID number:
Student Last Name:

Exercise 1. We recall below some logic gates:



ab	c
00	0
01	0
10	0
11	1

ab	c
00	0
01	1
10	1
11	1

a	b
0	1
1	0

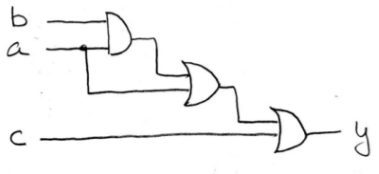
Consider the following truth table:

a	b	c	y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Give a gate diagram implementing the above truth table.

See Slide 14, PDF version of 5.3.

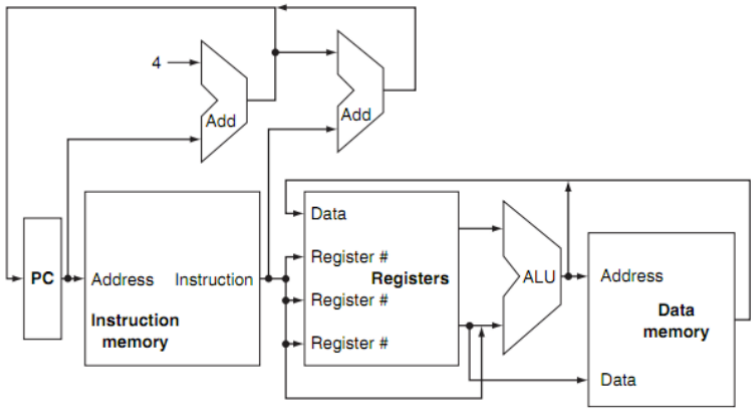
Exercise 2. Consider the following gate diagram



Write a Boolean expression equivalent to this gate diagram. Then simplify this Boolean expression and deduce a gate diagram equivalent to the above one.

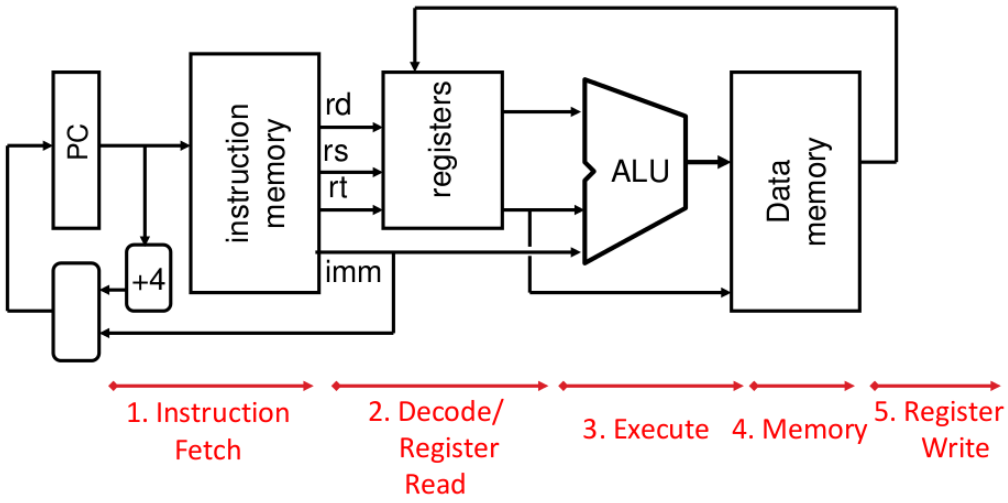
$$\begin{aligned} y &= ((a \cdot b) + a) + c \\ &= a \cdot b + a + c \\ &= a + c \end{aligned}$$

Exercise 3. Recall briefly the role of a multiplexer.
How many muxes do we need to add in the following diagram?
Explain briefly why.



The solution in on Slide 24 from the set of slides 5.5 (PDF version).
The mux is used when selecting from multiple inputs to the output.

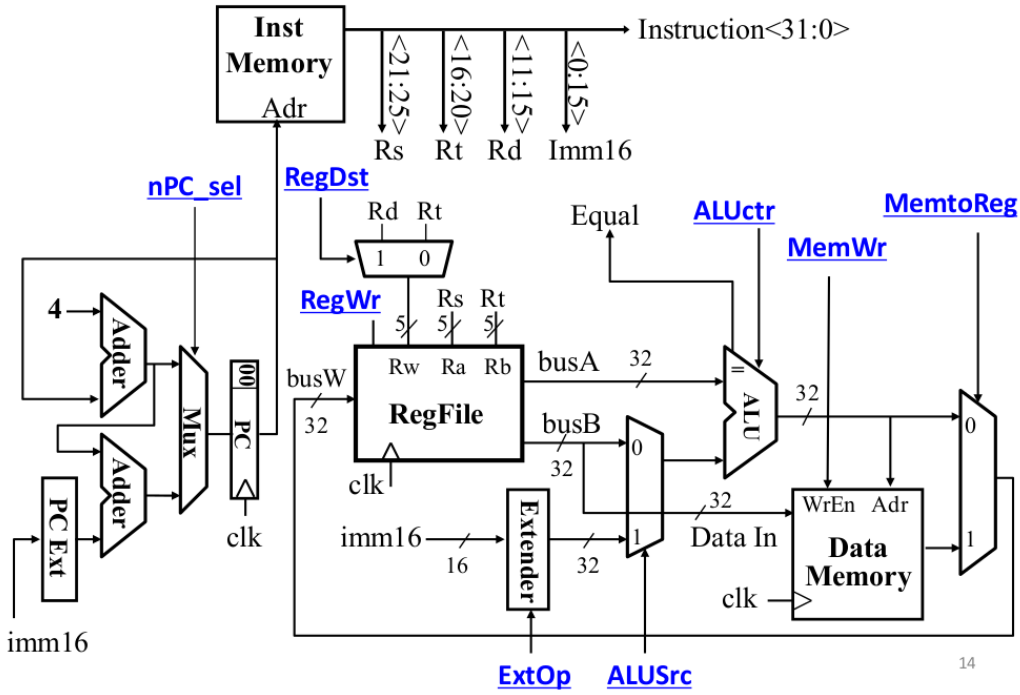
Exercise 4. Use the following sketch of a MIPS instruction single cycle data path to recall the data path of the `sw` (store word) MIPS instruction.



In addition, use the answer box to recall briefly in plain English what's happening at each of the five stages.

Slides 18 and 19 from the PDF version of 5.5.

Exercise 5. Recall the generic MIPS single cycle data path with control signals.



For the same `sw` (store word) MIPS instruction, give of the value of each of the 8 control signals.

nPC_sel = 0
RegDst = x / 0
RegWr = 0
ExtOp = 1
ALUsrc = 1
ALUctr = add
MemWr = 1
MemtoReg = x

Exercise 6. Consider a pipe-lined process (like the laundry example given in class) with s stages having equal duration.

If n tasks are being processed by the pipe-lined process,

1. what is the speedup w.r.t a serial execution
2. what is the percentage of time during which the pipeline runs at full occupancy?

1. The pipelined process takes $n - 1 + s$ clock cycles. For a serial execution, it takes $n \times s$ clock cycles. Thus, the speedup is $(n \times s)/(n+s-1)$. For a fixed s, when n escapes to infinity, the above ratio is asymptotically equivalent to $(n \times s)/n$, that is, s.

2. First, note that full occupancy happens at a given point of time whenver s tasks are using the pipeline at this point. Hence, full occupancy requires $n \geq s$. Now observe that, since all stages are equal in duration, the number of cycles during which the pipeline runs at full occupancy is equal to the total number of its cycles, that is, $s + n - 1$ minus the *fill up* and *drain out* times, that is, $2(s - 1)$. Thus the full occupancy time is

$$s + n - 1 - 2(s - 1) \;=\; n - s + 1.$$

   Therefore, the percentage of time during which the pipeline runs at full occupancy is

$$\frac{n - s + 1}{s + n - 1}$$

   which is asymptotically equivalent to 1 when n escapes to infinity.