Solutions

2.1

- PC, IM, RF, ALU, DM (then write-back to RF) ✓.

2.2

- RegWrite 0, MemWrite 1, MemRead 0 ✓.

2.3

- Either value leaves the instruction's visible behaviour unchanged ✓.

3.1

- `lw`: $200 + 100 + 200 + 250 + 150 = 900$ ps; cycle = 900 ps ✓.
- `add`: $200 + 100 + 200 + 150 = 650$ ps – under the cycle, so it waits; only the longest path sets the clock ✓.
- 100 instructions: $100 \times 900 = 90\,000$ ps ✓.
- DM 150 ps: `lw` path = 800 ps – now the new cycle; total 80 000 ps ✓.

3.2

- `lw R2, 0(R1)`: RegWrite 1, MemRead 1, MemWrite 0, MemToReg 1, ALUSrc 1, Branch 0 – data memory feeds write-back ✓.
- `sw R3, 4(R1)`: RegWrite 0, MemWrite 1, MemRead 0, MemToReg x, ALUSrc 1, Branch 0 ✓.
- `beq R1, R0, next`: RegWrite 0, MemRead 0, MemWrite 0, ALUSrc 0, Branch 1 – the ALU compares, memory untouched ✓.
- `add R4, R1, R2`: RegWrite 1, MemRead 0, MemWrite 0, MemToReg 0 (ALU result), ALUSrc 0, Branch 0 ✓.
- Versus `lw`: Branch $0 \to 1$, ALUSrc $1 \to 0$, MemToReg x; not taken means the PC simply takes PC + 4 ✓.

exercises.lol exercises.com.cn