תורגול כיתה מס' 10
Pipeline MIPS
הסבירו איך.Contracts (bypass/forwarding) הדם בパイプライン כדי לבצע את התכנית הבאה:

add $2, $3, $4
add $4, $5, $6
add $5, $3, $4
add $2, $3, $4
add $4, $5, $6
add $5, $3, $4
Warning: Reading old $4

No forwarding needed.
Rs=3 ≠ Mem: Dst=4
Rs=3 ≠ Wb: Dst=2
Rt=4 = Mem: Dst=4
Rt=4 ≠ Wb: Dst=2
הסברו איזה מעקף (bypass/forwarding)ควร ב-pipeline כדי לבצע את התכנית הבאה:

add $2, $5, $4
add $4, $2, $5
Sw $5, 100($2)
add $3, $2, $4
add $2, $5, $4
add $4, $2, $5
sw $5, 100($2)
add $3, $2, $4
Register File Split

- Register file is written during first half of the cycle
- Register file is read during second half of the cycle
  ⇒ Register file is written before it is read ⇒ returns the correct data
Hazard detection unit

```assembly
main: lw $t0, 0($s0)
    add $s1, $t0, $t1
    sub $s2, $t0, $t1
```

- Forward unit can not resolve the load hazard conflict.
- Bubble insertion is needed.
Clk 3 – hazard detection unit

```assembly
sub $s2, $t0, $t1
add $s1, $t0, $t1
lw $t0, 0($s0)
```
Don't Load the sub to the ID/IF reg. Keep the add there for the next clk.

Don't Increment PC – keep read the sub Inst. From Memory.

Insert a bubble to exe stage – by zeroing the Control bits that moves to the next pipe stage.

Don't Load the sub to the ID/IF reg. Keep the add there for the next clk. Cycle.

Clk 3

Hazard detection unit

exe.rdest (=exe.rt)
exe.MemRead

Instruction memory

PC

Registers

IF/ID.RegisterRs
IF/ID.RegisterRt
IF/ID.RegisterRd

lw ($s0, 0, $s1)

add $s1, $t0, $t1

Address of:
sub $s2, $t0, $t1

lw ($t0, 0, ($s0))
חומר רקע