	Total Amount			2,28,000

(Page 1)

(Q1) For processor X_1 Cycle per Instructions.

X_1 CPI = 1 + 0.3 × 2 = 1.6

(0.3 → Branch Instruct", 2 → cycle stall)

X_2 → correct 90% prediction removes stall & Remaining 10% → 2 cycle stall.

X_2 CPI = 1 + 0.3 × (0.9 × 0 + 0.1 × 2) = 1.06

∴ Speed up = $\frac{X_1}{X_2} = \frac{1.6}{1.06}$ = 1.5094 ≅ 1.51%

(Q2) Stage Delays = 150, 120, 150, 160 and 140 ns

Register delay = 5ns

t_{clk} (clock (cycle) time) = max (stage delays) + t_{reg} = 160 + 5 = 165 ns

∴ For 100 independent instructions:

$T_{total} = (k - 1 + n)\, t_{clk}$ where k = 5 & n = 100

∴ for instruction 1 to complete in 5 stages 4 next instructions will be in pipeline in stages so 2nd complete instruction requires only one clock to complete

∴ Total Cycles = k + (n − 1) = (n + k − 1).

(k → cycles for 1st, (n − 1) → Remaining)

∴ Total time = (5 − 1 + 100) × 185 = 104 × 185

Total Time = 17160 ns

Q3) for Non-pipelined processor : 2.5GHz

∴ 1/2.5GHz = 0.4ns time per cycle

∴ T_{non} for instruction = 5 cycles × 0.4 = 2 ns/instruction

for Pipelined Processor :-

1/2GHz = 1/(2 × 10^9 Hz) = 0.5 × 10^-9 = 0.5 ns

Average Cycle Per Instruction CPI

= 1 + (0.30 × 0.05 × 50) + (0.10 × 0.50 × 2)

(0.30: memory instruction; 0.05: 5%; 50: stall; 0.10: Branch Instruction; 0.50: 50%; 2: stall)

= 1 + 0.75 + 0.10 = 1.85

$T_{pipelined}$ for instruction = 1.85 × 0.5 = 0.925 ns/instruction

∴ Speedup = 2.0 ns / 0.925 ns ≅ 2.16 times

Q4) Total PO stage time for 100 instructions.

= 40 × 3 + 35 × 2 + 25 × 1 = 215 cycles

∴ Total time :- IF, ID, OF (3) Before PO (215) & WB After (1).

∴ Total Cycle = 3 + 215 + 1 = 219 cycles

Q5) As 3rd Question 1/2.5GHz = 0.4 ns per cycle

T_{non} Instruction time = 0.4ns × 4 cycles = 1.6 ns for full instruction (Non pipelined processor)

for Pipelined 1/2GHz = 0.5 ns per cycle No stall, & 5 stages

T_{pipe} = 1 × 0.5 = 0.5 ns

∴ Speedup = 1.6ns / 0.5ns = 3.2 times

	Institute			
	Total Amount			2,28,000

Q6) For old pipeline T_{clk} = max latency = 2.2 ns ∴ clock time = 2.2 ns

Page 3

Before Branching in Ex stage there are 2 stages IF & ID which can stalls if sits idle if EX stalls

As other stage have 1 CPI therefore stall will be 2 cycle per Branch

∴ CPI for old pipeline = 1 + 0.2 × 2 → stall per instruction
(0.2 = Branch instruction)

= 1.4

∴ $T_{old\ pipeline}$ = 1.4 (CPI) × 2.2 ns (max latency) = 3.08 ns

New design ⇒ 8 stages (latency) 2.2/3 ns = 0.733 ns, 1 ns, 1 ns, 1 ns, 0.75 ns

$T_{new\ pipeline}$ = (max latency) = 1 ns → Time per CPI

Stages IF, ID, RF1, RF2, EX1, EX2, MEM, WB

5 stages can be idle if 6th stage EX2 fails to produce pointer for next instruction

As per stage have 1 CPI therefore 5 cycle stall per branch

∴ CPI for new pipeline = 1 + 0.2 × 5 → stall cycle
(0.2 = Branch instruction)

= 2.0 CPI

∴ $T_{new\ pipeline}$ = 2.0 (CPI) × 1 ns (1 time) = 2 ns

∴ Ratio P_1/P_2 = 3.08 / 2.0 = 1.54

Q7) for 6 stage no overhead ~~that means 6 stages in 1 cycle for pipe~~

Page 4

Now All 6 stages takes same time for execution.

25% instructions has ⟶ 2 stall cycles each.

We have to speed up this pipeline over non-pipelined design.

For Non pipeline design ⟶ 6 stages per instruction ∴ 6 cycles for instruction execution

∴ Time/instruction = $6 \times T$ ↳ where T = time per Cycle

If N instruction the $T_{Nonpipe} = N.6T$

Now Pipelined execution (No stall) → Ideal case.

We will complete 1 instruction per cycle

∴ Time/instruction = $1 \times T = T$

But our case has 25% instruction ⟶ 2 ~~stalls~~ cycle stall

∴ CPI ~~Time/instruction~~ = $1 + 0.25 \times \frac{2}{\text{stall cycle}}$ = 1.5 CPI
(↓ instruction)

∴ Time/instruction = $\frac{1.5 \times T}{\text{CPI}} = 1.5T$

∴ For N instruction Tpipeline = $N.1.5T$

∴ Speed up = $\frac{T_{Nonpipe}}{T_{pipeline}} = \frac{N.6T}{N.1.5T} = 4$

Q8) Note for Non-pipelined design

Total Cycle time = Sum of stage delay + one Reg delay

= (5+6+11+8) + 1 = 31 ns for one instruction

But For pipelined Design.

Tpipe Cycle time = Max stage delay + reg delay

= 11 + 1 = 12 ns

∴ Speed up = $\frac{\text{Total } T_{non}}{T_{pipe}} = \frac{31}{12} = 2.583 \simeq 2.6$

	Total Amount		2,28,000

Page 5

Q9) IF → ID → EX → WB

IF: 1 cycle; ID: 1 cycle; EX: ADD & SUB → 1 cycle, MUL → 3 cycle; WB: 1 cycle

Thus MUL instruction causes → stall of 2 cycle

∴ Total Time

C1 → I1 → IF

C2 → I2 ID | I1 IF

C3 → I3 IF | I2 ID | I1 EX (Add)

C4 → I4 IF | I3 ID | I2 EX (MUL) | I1 (WB)

C5 & C6 → Stall

C7 → I2 (WB)

C8 → I3 (WB)

4 cycle for 1 instruction for SUB

+ 2 cycles for other instructions

+ 2 cycle stall

= 8 cycle total for 3 instruction.

Q10) Five stages for 1 instruction takes 1 cycle

Now clock rate 1 GHz thus cycle time = $\frac{1}{1\times10^9 Hz}$ = 1 ns

Conditional Branches = 20% & Branch Decision in Stage 3

Therefore 2 stages stalls for Branching & each stage 1 cycle thus Stall = 2 cycle for Branch.

∴ Average CPI = 1 + 0.2 × 2 = 1.4 (Branch instn → stall)

∴ Time for 10^9 instruction = 1.4 CPI × 10^9 instn × 1 ns cycle time = 1.4×10^9 ns = 1.4 sec

for 10^9 instruction = 1.4×10^9 instructions $\times$ 1 ns (cycle time/instruction)

$= 1.4 \times 10^9$ ns

$= 1.4$ sec

(Q11) 5 stages IF, RD, EX, MA, WB. 4 cycle each stage.

For Instruction 1 still 4 stages for I2. I2 cannot enter as R0 required in I2 from I1 only after WB stage in I1, it is usable in I2.

∴ At 5th Cycle I2 IF stage enter. I1 will complete WB stage completion.

Same I2 instruction will leave in cycle 9 for WB stage I3 will enter in IF stage.

R0 updated in WB stage of I2 will be used in I3 stage RD in Cycle 10

~~Expl~~ R3 will complete in Cycle 13

∴ No of Cycles Required = 13

(Q12) offset = Target − (address of next instruction)

Each instruction = 4 Bytes. offset is in bytes

Target of Branch instruction is i

Suppose i has base address A

i+1 will be at address A+4

i+2 — " — A+8

i+3 — " — A+12

i+4 (next after Branch) at add A+16

Target Addr = Addr of Next Inst^n + offset.

Target Address = Address of inst^n i = A.

Addr of next inst^n = Address of i+4 = A+16

∴ A = (A+16) + offset

∴ offset = A − (A+16) = −16

12) 4 stage. delays 150, 120, 160, & 140 ns. Reg delay 5ns

n = 1000 data items

Time per cycle = Max delay + Reg delay = 160 + 5 = 165 ns

∴ for 1000 data items

Total cycle = $n + (k-1) = 1000 + (4-1) = 1003$ cycles

∴ Total Time for 1000 = 1003×165 ns = 165495 ns

Answer: -16

Q13: Consider a pipeline consist of 5 stages named as IF, ID, OF, EX and WB with the respective stage delays of 2 ns, 6 ns, 5 ns, 8 ns and 1 ns. The alternative pipeline 'y' contain the same number of stages but EX stage is divided into 2 substages, (EX1 and EX2) with equal delay i.e. (8 ns/2) and ID stage is divided into 3 substages (ID1, ID2 and ID3) with equal delays of (6 ns/3). In the pipeline x and y memory reference instructions are not overlapped so the penalty of memory reference instructions in the pipeline is 4 cycles and in the pipeline 'y' is 8 cycles. If the program contain 20% of the instructions which are memory based instructions, what is the ratio of speed-up of x to speed-up of y?

→ for pipeline x → cycle time t_x = max = 8ns for stage EX.

y → cycle time t_y = max = 5ns for stage OF

Memory ref penalty = x = 4 cycle/instruction, y = 8 cycle/instruction, 20% memory instr → $p = 0.2$

Cycle per Instruct^n (CPI) with stalls $CPI_x = 1 + p \times 4 = 1 + 0.2 \times 4 = 1.8$

$CPI_y = 1 + p \times 8 = 1 + 0.2 \times 8 = 2.6$

Time per instruct^n:- $T_x = CPI_x \times t_x = 1.8 \times 8 = 14.4$ ns

$T_y = 2.6 \times 5 = 13.0$ ns.

Ratio of speed up (x to y) $\frac{S_x}{S_y} = \frac{T_y}{T_x} = \frac{13.0}{14.4} \approx 0.903$

$S \propto \frac{1}{T_{avg}}$

speed up is inversely proportional to execution time

$\therefore S_x : S_y \approx 0.903 : 1$ (pipeline y is ≈11% faster.