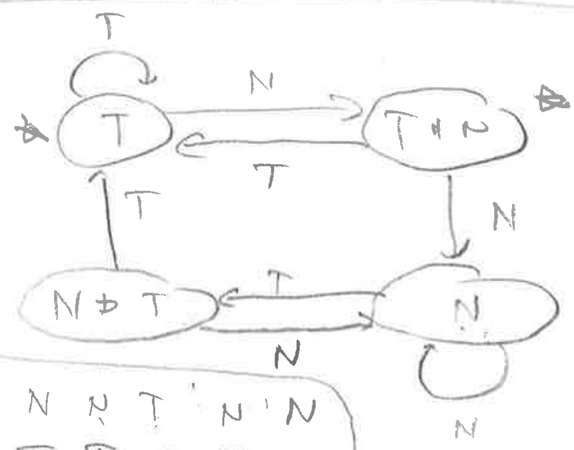
Draw the 2-bit branch prediction FSM. Then, given the 2-bit branch prediction method, with the initial state of N*, and the following set of branches, Describe the set of branch predictions

N, N, T, T, N, N, T, N, N

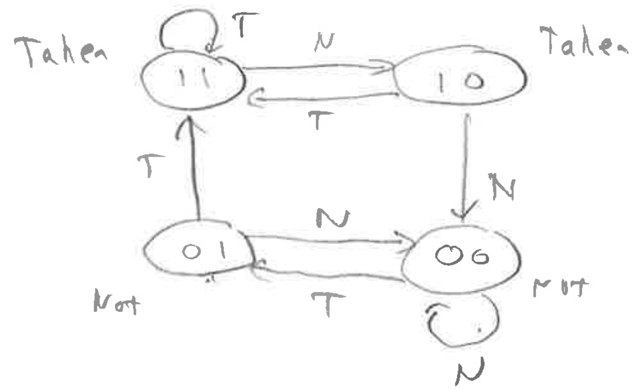


N*									
Actual	N	N	T	T	N	N	T	N	N
Predicted	N	N	N	N	T	T	N	N	N
Next	N	N	N*T	T	T*N	N	N*T	N	N

Draw the 2-bit branch prediction FSM*. Then, given the following branch addresses and branches, show the final state of a k=3 correlating prediction model

1	10001101	T
2	10001101	T
3	10001000	T
4	10001111	N
5	10001000	N
6	10001101	T
7	10001000	T
8	10001000	[illegible]
9	10001000	T

10001101 N



	LHT	LPT
0	000 → 100 → 010 → 101 → 110 → 111	00 → 01 → 11 → 10
1	000	00
2	000	00 → 01
3	000	00
4	000	00 → 01 → 00
5	000 → 100 → 110 → 111 → 011	00 → 01
6	000	00 → 01 → 11
7	000 → 000	00 → 00

Assume 40% of instructions change the flow of a program

- 16% of instructions are branches
  - 50% of branches are taken
  - Mispredicted branches result in a 3 cycle stall (wait for address to be calculated)
- 24% of instructions are loads
  - 50% of the time, the next instruction uses the loaded value

What is the impact on performance assuming:

- There is a 1 cycle stall for load hazard
- Always predict branch not taken
- 5-stage datapath

Branch 16% →

50% → 1 instr ⇒ $0.08N$

50% → 1 + 3 instr = 4 ⇒ $0.08(4N)$

Load 24%

50% → 1 instr ⇒ $0.12N$

50% → 1 + 1 stall = 2 ⇒ $0.12(2N)$

$(K+N-1)t_p \Rightarrow K=5$ stages $\Rightarrow (5+N-1)t_p = (4+N)t_p$

$N \Rightarrow (1-0.16-0.24)N + 0.08N + 0.08(4N) + 0.12N + 0.12(2N))$

$\Rightarrow \boxed{1.36N}$

$(4+1.36N)t_p$

Impact

$$\lim_{N\to\infty} \frac{(4+1.36N)\cancel{t_p}}{(4+N)\cancel{t_p}} = \boxed{1.36}$$