Q1: CS230 designed a 5-stage unpipelined processor that takes five ns per instruction. The inter-stage latches take 0.2 ns. CS330 created an equivalent 5-stage sequential pipeline with each stage taking the same amount of time. Assuming there are no stalls, what is the clock cycle time for the CS230 designs?
Ans1: 5ns

Explanation:

It is given that 5-stage unpipelined processor takes 5ns

For every 5ns one instruction get executed , so clock cycle time = 5ns

Q2: Refer previous question. Assuming there are no stalls, what is the clock cycle time for the CS330 designs?
Ans1: 1.2ns

Explanation:

In CS330 it is given that all stages take equal time , so each stage take 1ns

Inter-stage latches take 0.2ns

So for every 1.2ns one instruction get executed, so clock cycle time = 1.2ns

Q3: Refer previous question. How long does it take to finish one instruction in the unpipelined processor explained in the previous question?
Ans2: 5ns

Explanation:
It is specified in the question that unpipelined processor takes 5ns

Q4: Refer previous question. How long does it take to finish one instruction in the pipelined processor explained in the previous question?
Ans2: 6ns

Explanation :

There are total five stages and inter-stage latches take 0.2ns, each stages take 1ns

Total = 5 * 1ns + 5 * 0.2ns

So it takes 6ns for one instruction to finish

Q5: Which of the following statements are TRUE?
(a) Because of symmetric instruction formats with MIPS, it is easy to design a five-stage pipeline.
(b) Variable length ISAs complicate the decode stage of the pipeline.
Ans3: Both

Explanation:

a) Because of symmetric instructions, decoding logic becomes fairly simple to implement. And since the size is fixed, it can fetch pc+4 blindly no need to go to decode stage to get the size of instruction.

b) If its fixed length ISA we know the number of bits occupied by opcode so it is easy to decode. But in variable length we don’t know exact number of bits used by opcode, so decoding is complex.