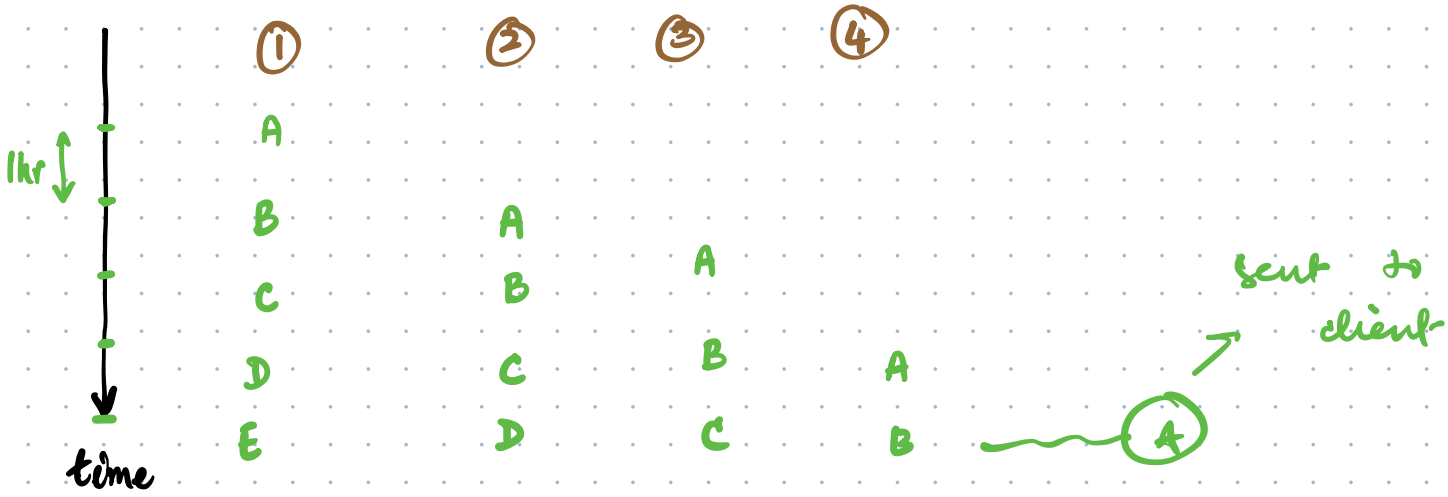
April 8, 2026

Optimization

- it takes 4 hours to assemble a car
- throughput: 0.25 cars per hour
- latency: 4 hours

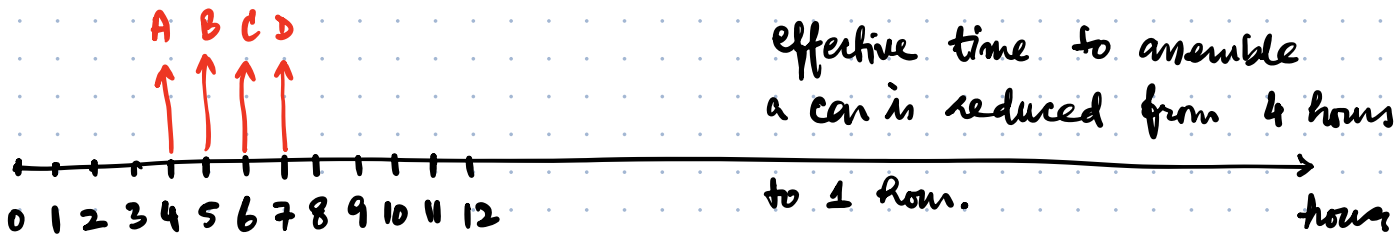
Goal: the goal of optimization is to reduce latency ↓ and increase throughput ↑

* Pipelining



Observation: we are able to produce cars at the 1hour mark.

effective time to assemble a car is reduced from 4 hours to 1 hour.

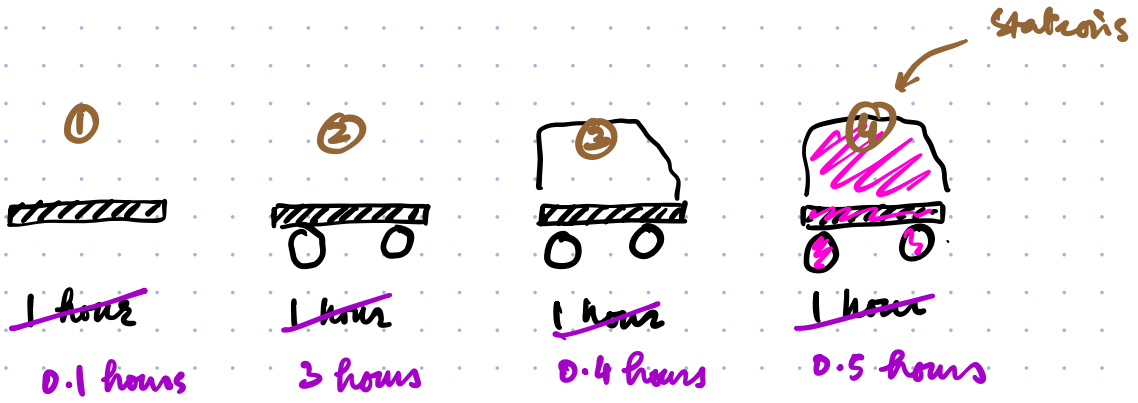


- Using pipelining we have increased the throughput to 1 car per hour.
- 4x more efficient at producing cars.

Q. What if the pipeline has n stages?

A. Throughput will be increased n times.

Throughput pipeline is n stages = n x Throughput w/o pipelining



- Unbalanced stages

bottleneck. only as fast as the slowest stage.

- we can only produce a car every 3 hours.

∴ Throughput = 0.33 cars per hour

* Pipelining is an implementation technique where multiple instructions are executed in parallel.

* Instruction: ① Fetch ② Decode ③ Execute

- Many situations where pipelining becomes challanges;
  - (i) exceptions
  - (ii) interrupts
  - (iii) faults
  - (vi) jumps

- Synchronous events ← internal to the program
- Asynchronous events ← external events

- User requested tasks ← easier, can handle after an instruction is completed
- Coerced tasks

↑ difficult to predict
caused by hardware

* Threats to pipelining.

(i) data dependence : use the same data

(ii) name dependence : access the same memory or hardware registers.

(iii) control dependence : order of instruction w.r.t. branches.

* Multicore CPUs
  - Multiple physical units: ALUs, control units, register sets
* GPUs
  - Multiple cores