Parallelism vs. Concurrency:

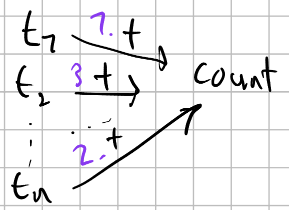
Parallelism: (mostly part 1 of the course)

"Doing mupltiple things at the same time": try to speedup execution of a program with threads

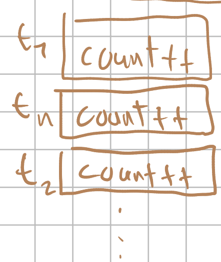
example: Sum an array

Only makes sense/possible if a large part of the program is "parallelizable":

e.g. incrementing a counter like in assignment 3 yields no benefit



→ increments have to be sequential anyway



Note: we have no guarantee that things will actually be parallel !

Depends on how nice the scheduler is. Also possible that our program just gets scheduled more often if we give the OS more threads.

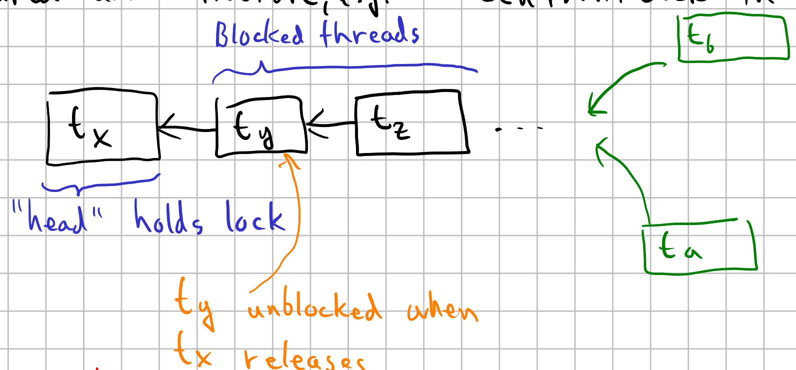
Concurrency: (mostly part 2)

"Dealing with multiple things at the same time": We are in a situation where multiple threads can access a shared resource → Manage access to ensure correct behavior.

example:

① shared counter

② shared data structure, e.g. "ReentrantLock" in Java



concurrent enqueues
→ how to deal with this
+ make it fast?

↳ this course will help you understand a data structure like this.

Library:

books A, B, C, D: A [2], B [1], C [3], D [1] — 1 copy of each!

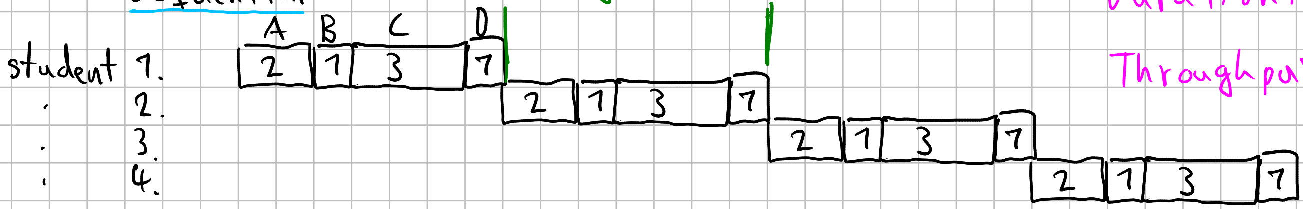
↓ 2 time units to read

Latency? Balanced? Duration? Throughput?

Students must read A then B, then C, then D.

Sequential:

latency of student 2



Latency: 7 → same for all → balanced

Duration: 4·7

What is throughput?

"Throughput measures the amount of work (e.g. CPU instructions) that can be done by a pipeline in a given period of time."

→ #items (e.g. #students)

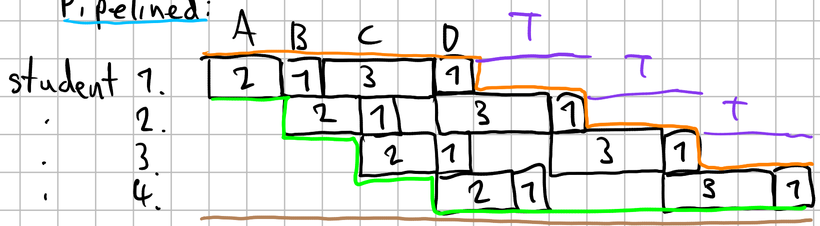
→ unit: $\frac{\text{items}}{\text{time}}$

Analogous to velocity: "how many meters travelled per second?"

x meters in y seconds → velocity $= \frac{x}{y} \frac{m}{s}$

$\Rightarrow$ Throughput $= \frac{\#\text{elements}}{\text{time}} = \frac{4}{4 \cdot 7} = \frac{1}{7} \frac{\text{students}}{h}$

Pipelined:



→ latency 7
→ latency 8 ⇒ unbalanced

Latency: (in this course) not defined since it differs for each element.

Duration: different ways to compute: (brown) (green) → need diagram

(orange) → if we know T we can compute it as $7 + (4-1) \cdot T = 16$

Throughput? two definitions

ⓐ (as before) $\frac{n}{\text{duration}} = \frac{4}{16} = \frac{1}{4}$

Now consider $n \to \infty$:

$$\frac{n}{\text{duration}} = \frac{n}{L + (n-1) \cdot T} = \frac{n}{(L-T) + n \cdot T} = \frac{1}{\frac{L-T}{n} + T} \xrightarrow{n \to \infty} \frac{1}{T}$$

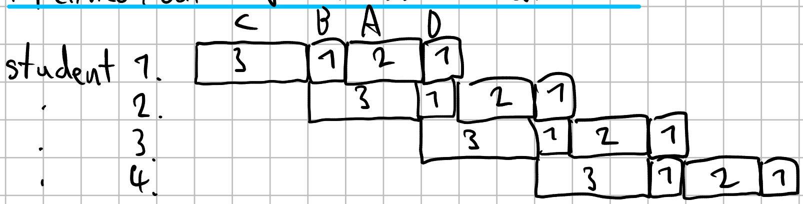
↳ sum of length of all stages

T := time between two elements in consecutive iterations finishing

Should always be constant in a pipeline with one "execution unit" per stage (e.g. 1 copy of each book)

⑥ $\frac{1}{T}$ → I would use this def., both are correct for the exam

Pipelined, but stages A and C switched:



→ balanced ☺

In general, in a pipeline with one "executio unit" per stage:

pipeline balanced ⇔ no stage is longer than the first and

T = length of the longest stage (⇒ duration = L + (n − 1)·T)

L ↓ sum of length of all stages