Ask me anything

Mentimeter

9 questions
6 upvotes

What experience do you have with concurrent programming?

Mentimeter

None

Concurrency Lecture @ unine

Actor based programming with akka/java

Experience from a course

Hello, I am pretty new to this concept

Working as Java-Developer for 4y

Almost none

not much :)

Shallow experience in Java; from job and personal projects

21

What experience do you have with concurrent programming?

Mentimeter

I've worked with concurrency with the Go programming language.

course in operating systems & scheduling, simultaneous database querying, threading in java

Message Passing Interface

Scala course

Working and education as a software engineer for 10+ years in java and scala mainly. So mostly I'm familiar with functional programming parallelism

I used it only in a C course. Pipes, Child/Parent processes and Fork().

none

Tried various technics from a book about Java9.Parallelizing physically based ray tracer, needing to synchronize on some counters/progress coutners. (written in Rust)

Probably used, hidden in the (UI) framework

21

What experience do you have with concurrent programming?

Mentimeter

Not really familiar

3, 2, 1

so it's always ready for user input and never busy with another task

21

What are possible values of x after P1 and P2 complete?

Mentimeter

```
  { x = 0 }
P1: x := x+1
P2: x := x+2
  { x = ? }
```

1,2,3

1,2,3

1, 2 and 3

1,2,3

1, 2, 3

1, 2, 3

1,2,3 depending on P1 or P2 write first

Threads are the smallest unit that can be executed on a processor, right?

8

What's the difference between a process and a thread?

Mentimeter

a process can have multiple threads of control

process can have multiple threads. Thread don't execute in parralel

These terms are usually used interchangeably but I think in this context, a process can have several threads. Process has its own memory space but threads share memory space

process typically has it's own address space, while multiple threads of a process typically share the address space

File Descriptors are on Process Level

Threads are lightweight

Normally processes are considered to be standalone software on the OS level with their own address space. Threads are within the same running software

Threads use the same memory address space

For interactivity

9

Why is a GUI always multi-threaded?

Mentimeter

For interactivity

It has to listen for user input.

To handle different input devices responsively

It needs to read from IO and display at the same time

a running task should not prevent the user from interacting with the UI

Because a GUI usually is waiting for input from (multiple) input sources/devices

Because it needs to listen to input events while the program maybe doing something else.

To ensure responsiveness

Otherwise the user interaction won't be responsive

13

Why is a GUI always multi-threaded?

Mentimeter

it needs to listen to different kinds of user input

async interaction

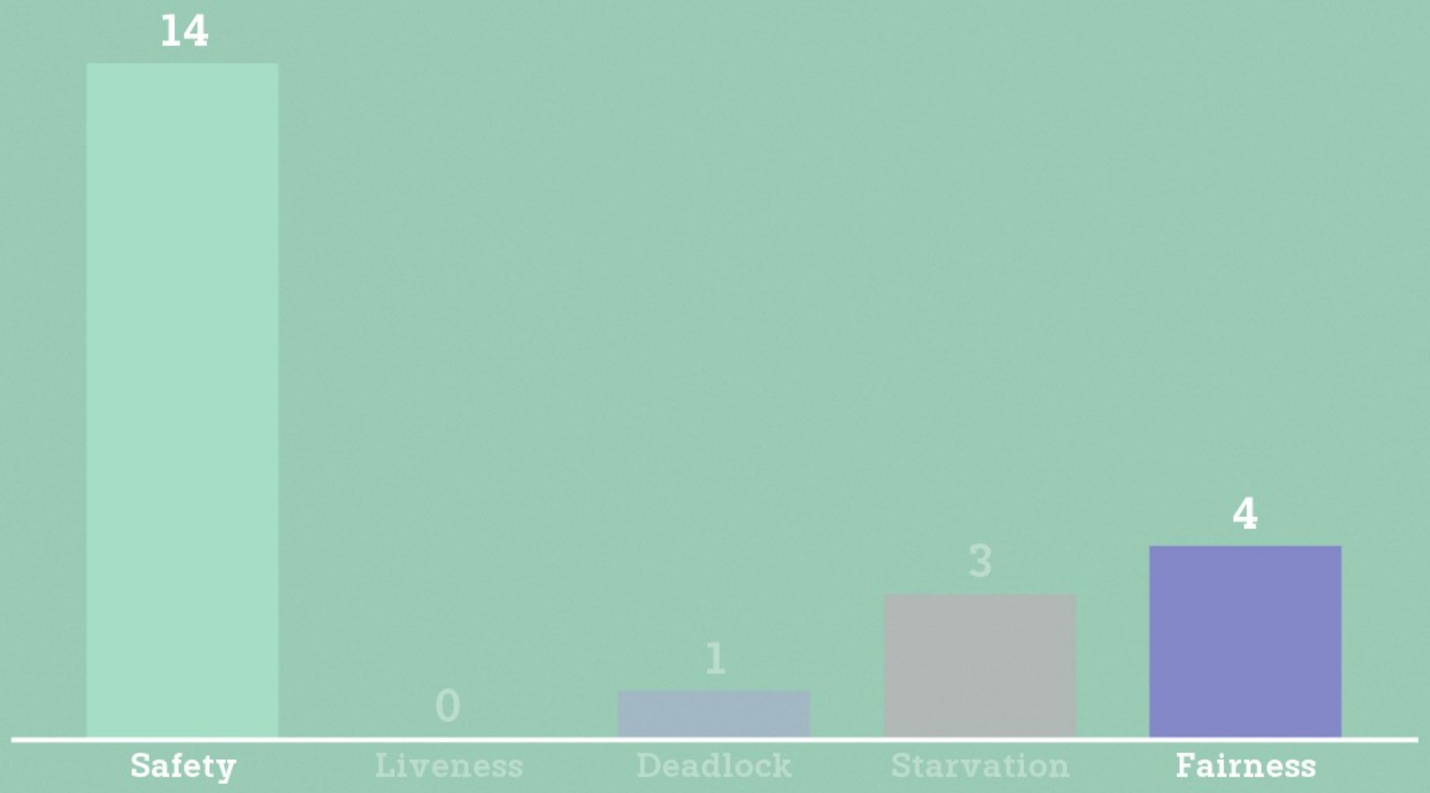
Because in GUI delay is important and therefore, it needs to do several tasks concurrently (such as receiving data from the user, outputting a response, etc.)

because the useraction that happens on the gui and it's rendering needs to be done seperatly from what is done with the data in the backend

13

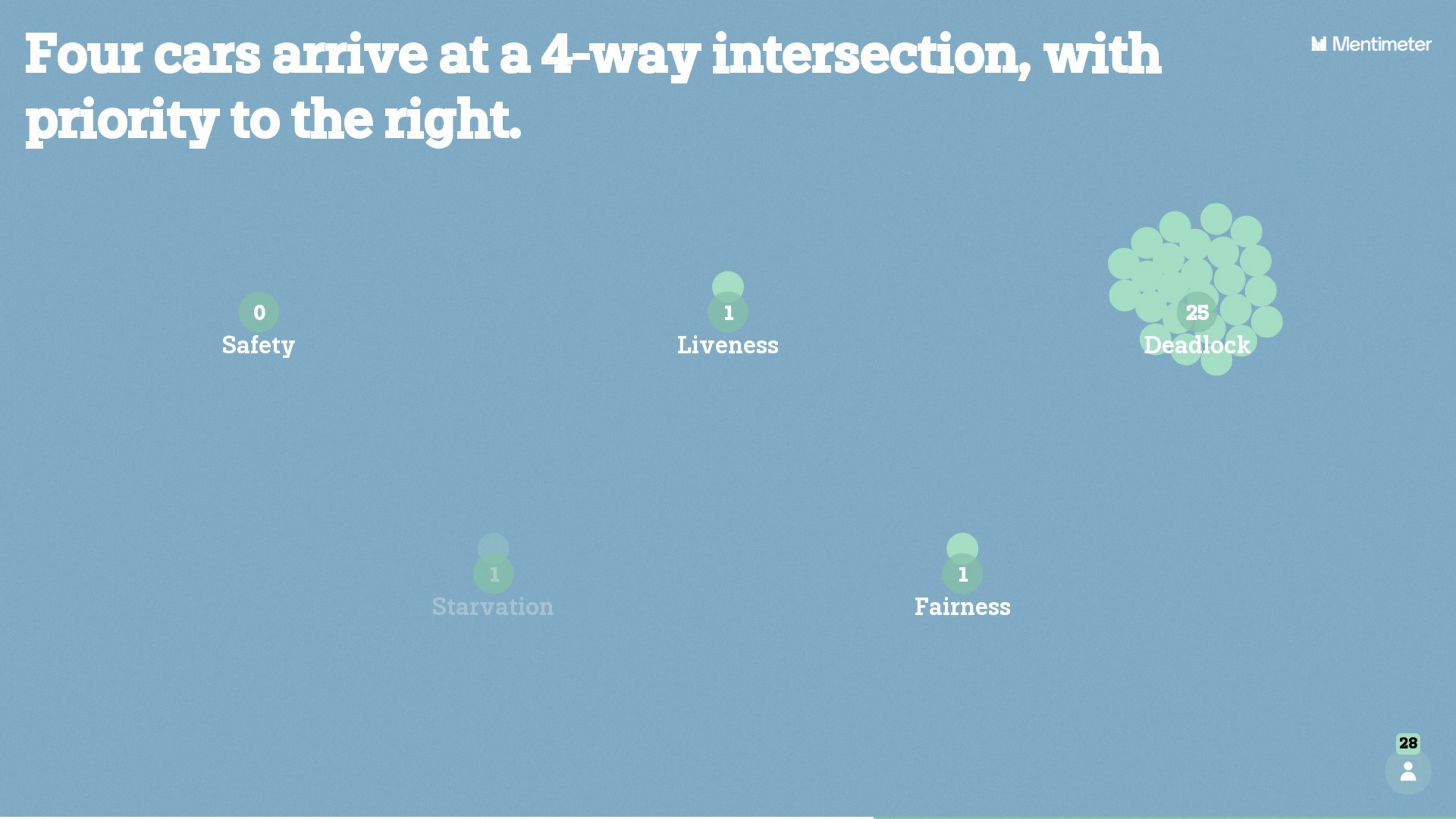
Two people attempt to withdraw money from the same account at the same time.

Mentimeter

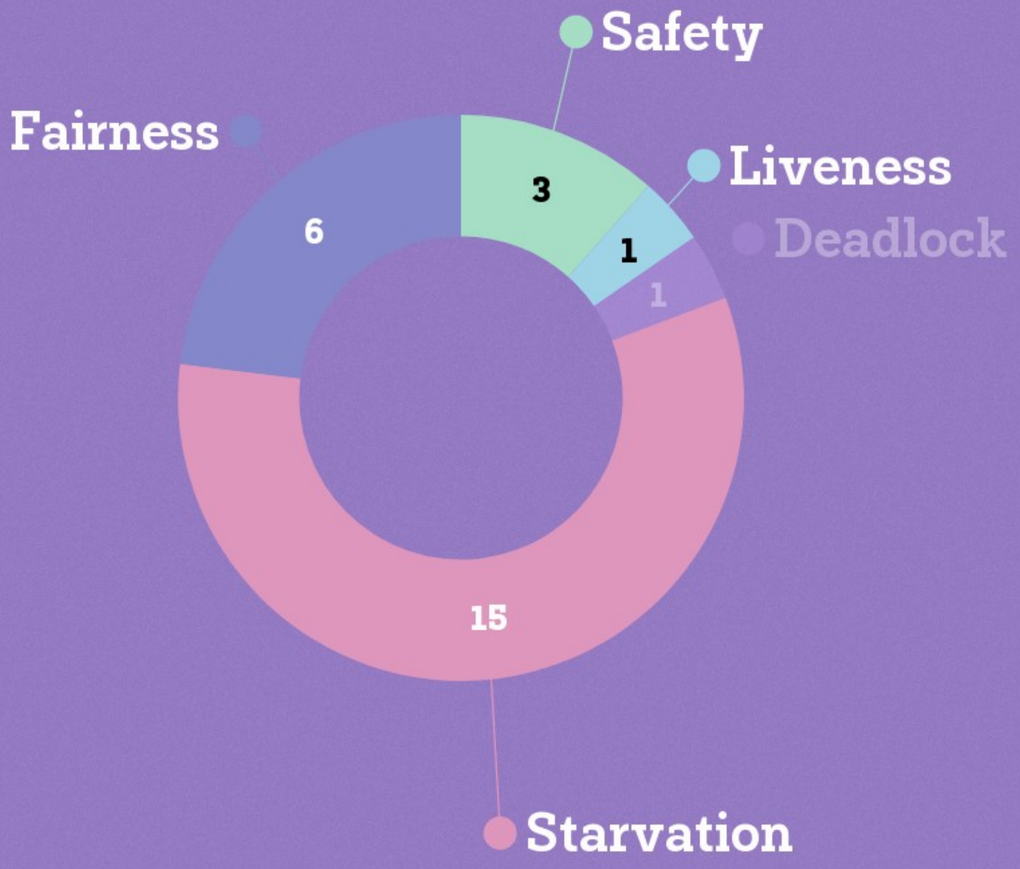


22

Four cars arrive at a 4-way intersection, with priority to the right.



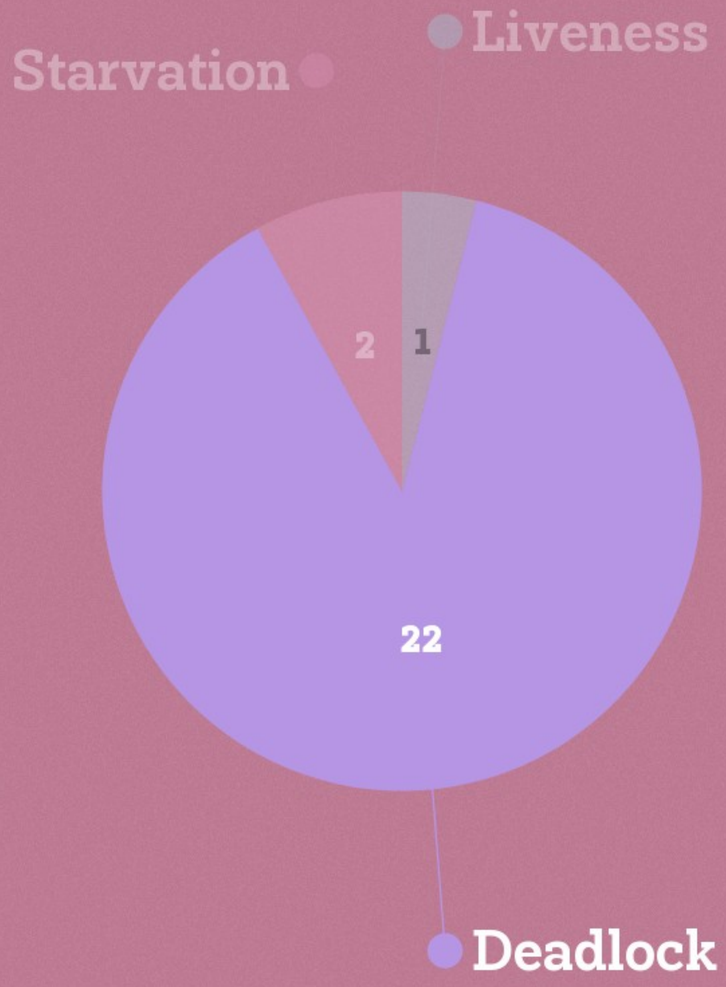
A car is in a roundabout, and another car wants to enter.



26

I need a bank account to get a work permit, but I need a work permit to get a bank account.

Mentimeter



25

Which should be easier to program with: semaphores or monitors? Why?

Mentimeter

without a monitor i can't see what i'm programming

Semaphores because they have an easier logic. With monitors one should be careful regarding recursive calls and etc.

monitors, because they're more high level

monitors because they offer more "oop functionalities"

Depends but in general I'd say monitors since they're richer in structure. A semaphore could be easily implemented in a monitor by the monitor being of size 1, holding a unit value

semaphores

Monitors is easier (as the releasing part is taken care of)

Semaphores, since it is easier to understand.

(Depends strongly on language/vm/compiler)

11

Which should be easier to program with: semaphores or monitors? Why?

Mentimeter

Semaphores

i have only learned about semaphores, so far. So, we'll see I guess during the semester.

11

Mentimeter

Last chance for questions