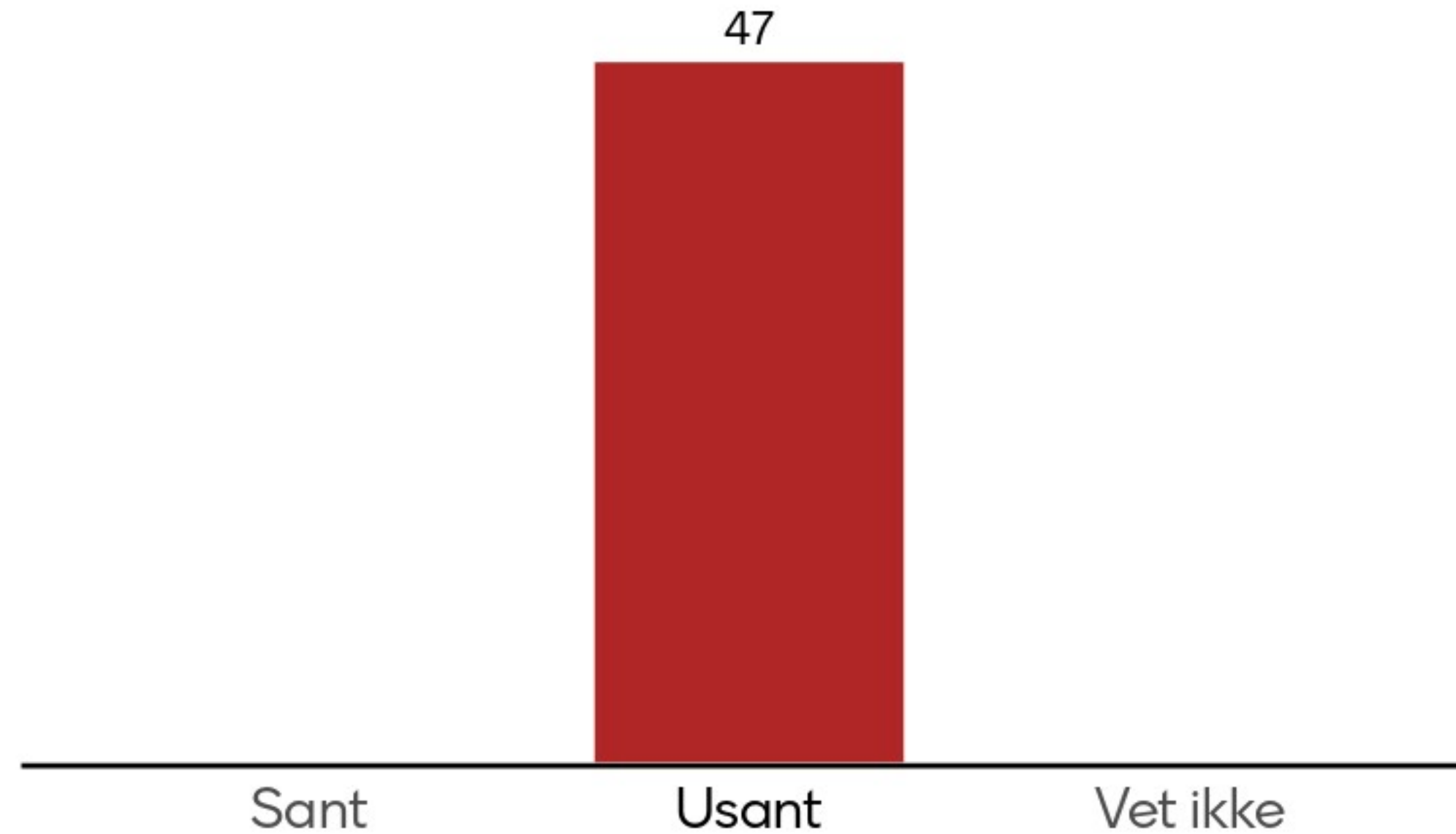


IN1010 Repetisjonstime - Tråder

Agenda

- Menti-quiz for å se status
- Gå gjennom prinsipper
- Liveprogrammering
- Oppgaver, QnA, avklaringer

Tråder gjør at programmet alltid kjører raskere



Brukes for å beskrive monitor-mønsteret...

kokker og
servitører

produsenter
og
konsumenter

signalisering
og venting

Låsing, venting og signalisering (gjennom Conditions) foregår kun i monitorene

Sant

Usant

Vet ikke

Brukes for å sørge for at kun én tråd om gangen kan bruke ressursen...

Låser

Unntak

Try-catch

CountDownLatch

Ved å kalle på `Thread.sleep()` kan det kastes...

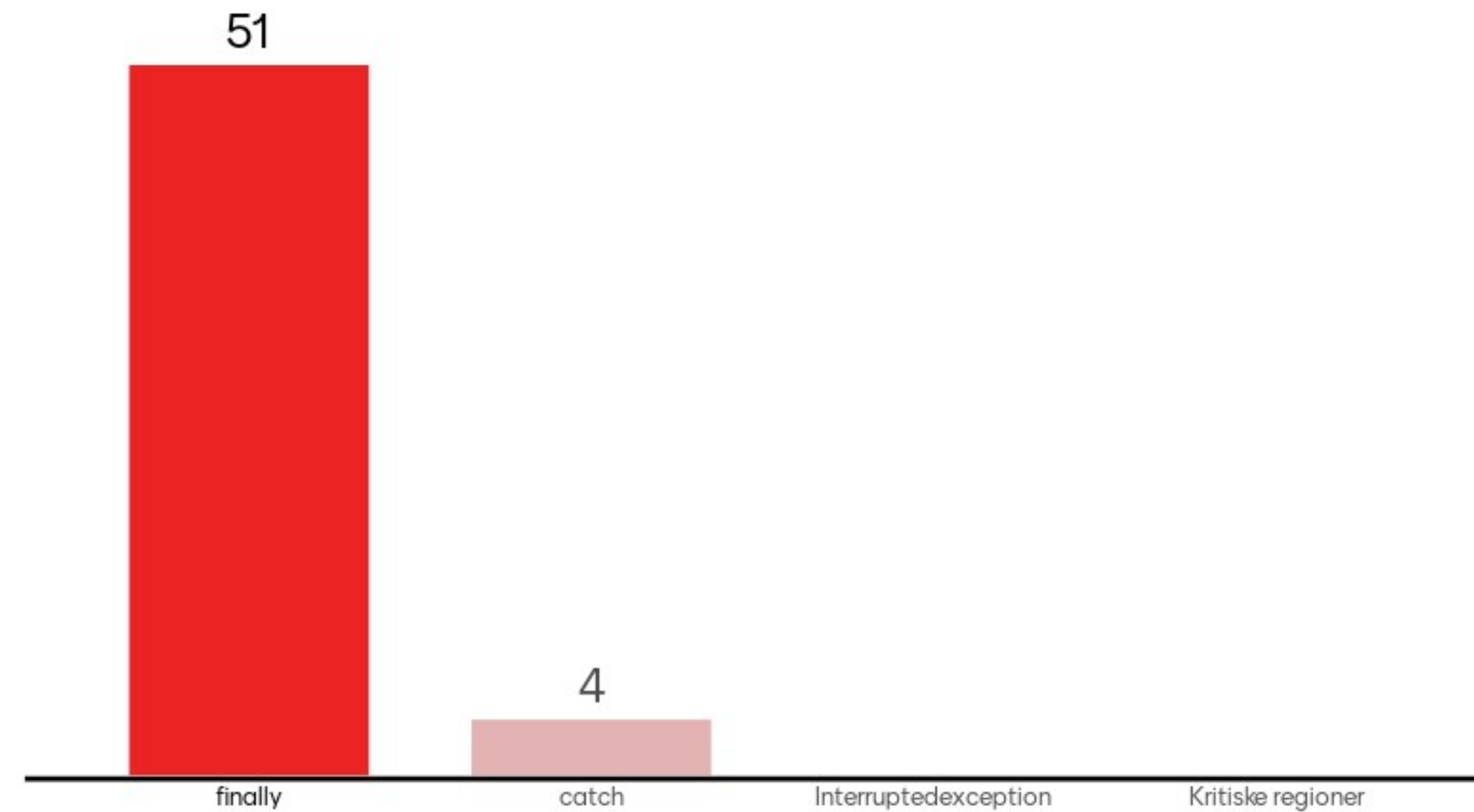
`NullPointerException`

`NoSuchThreadException`

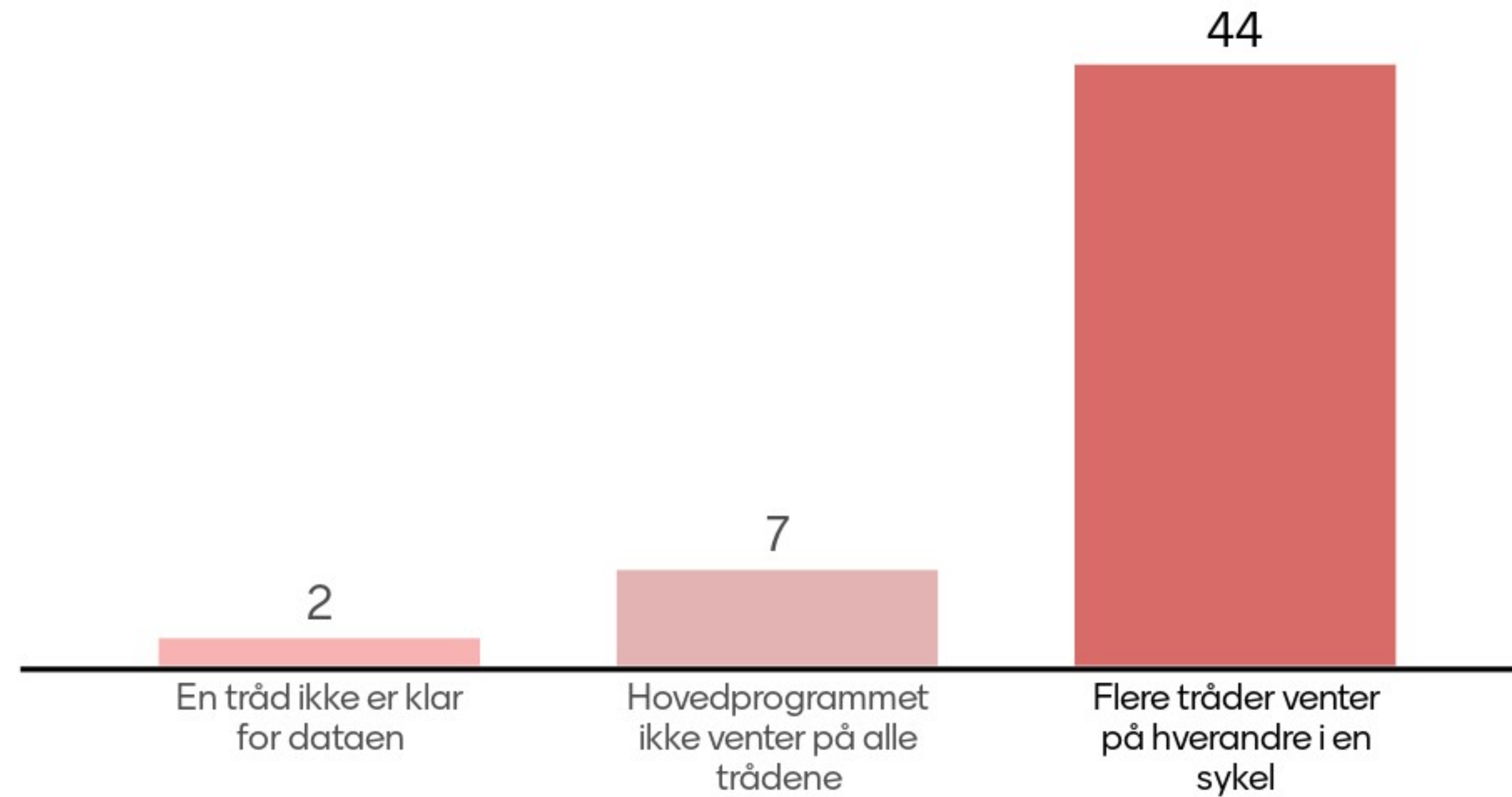
`InterruptedException`

`NoSuchElementException`

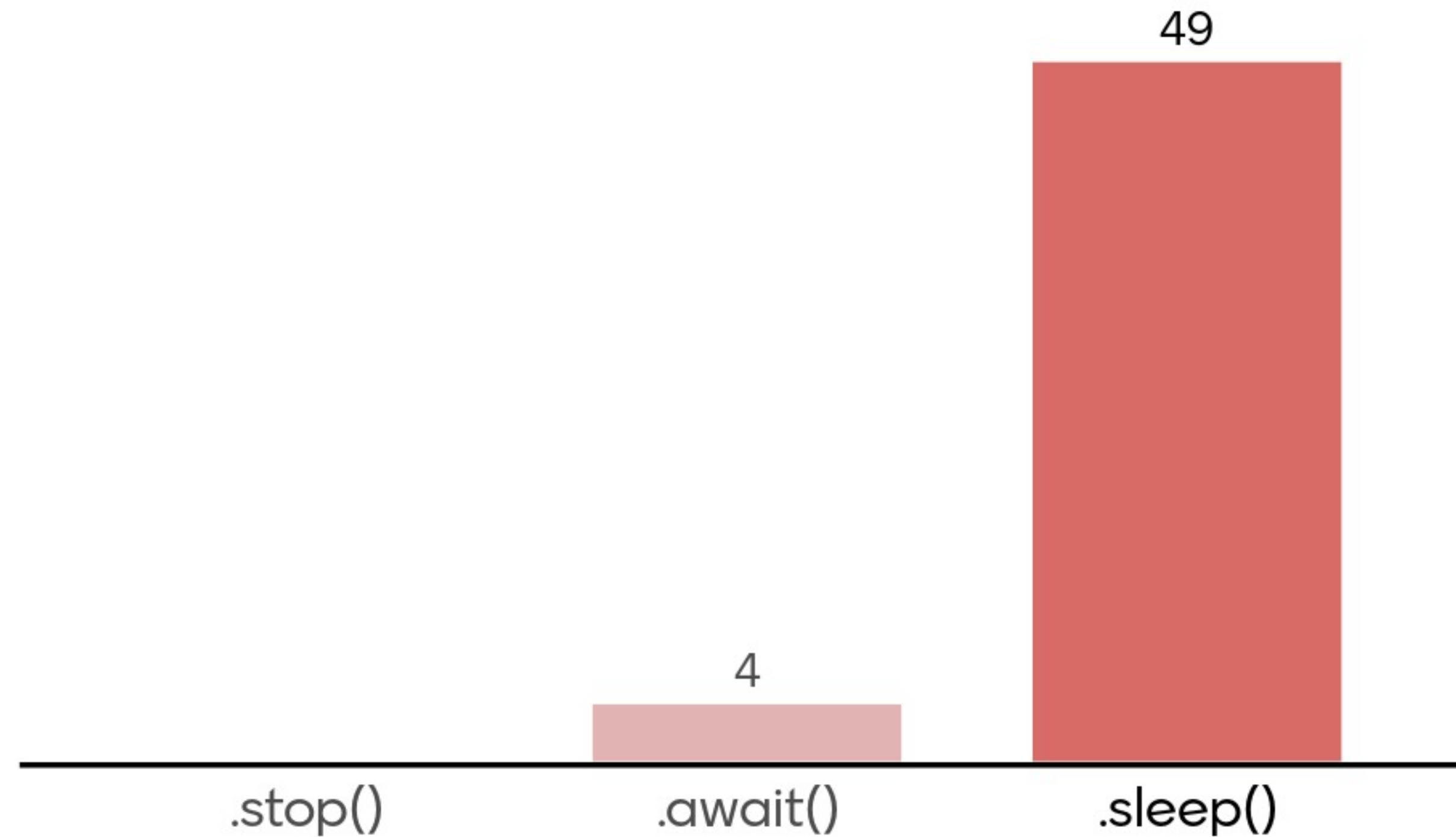
For å sørge for at en tråd alltid gir slipp på låsen bruker vi...



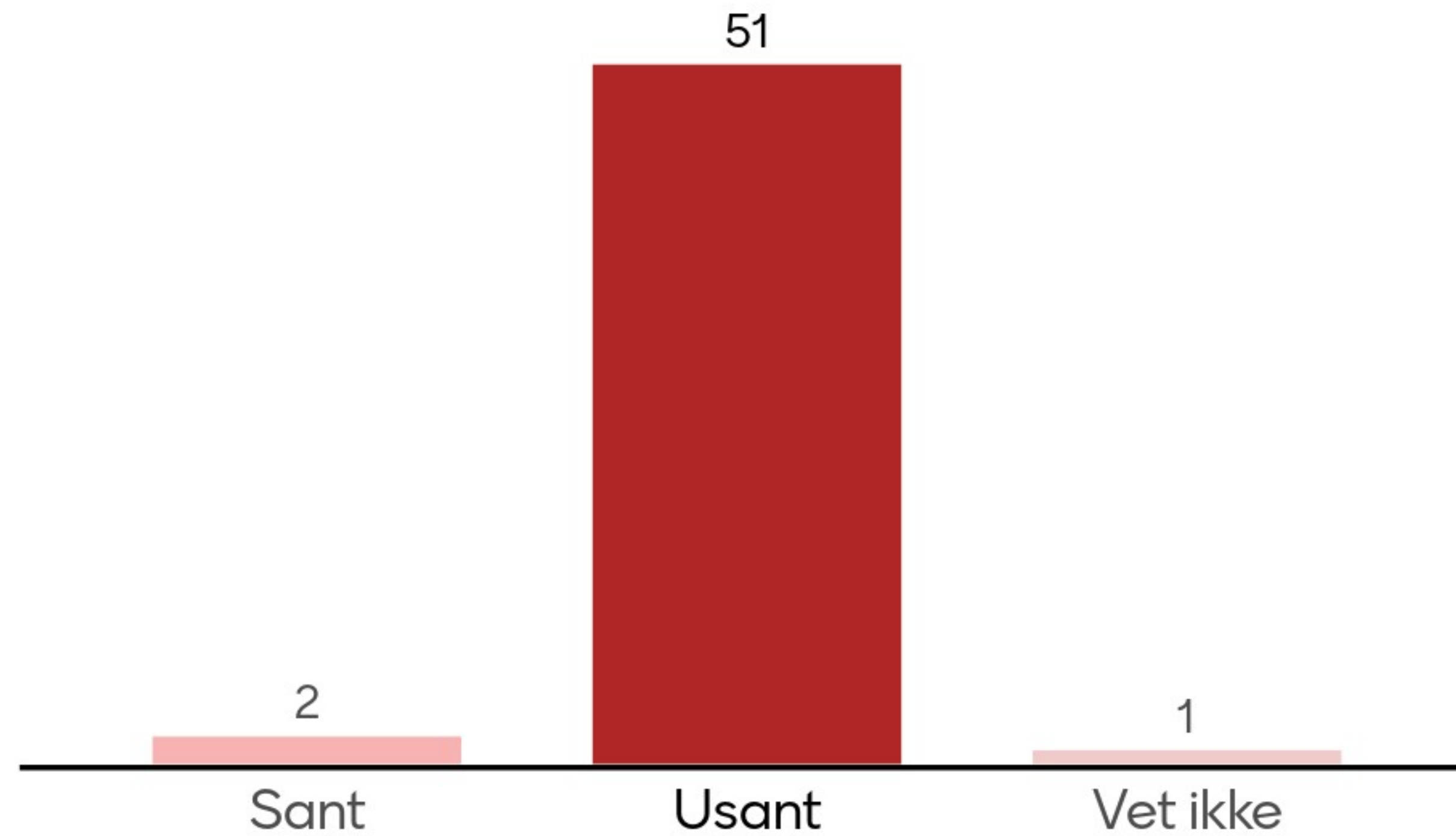
Vranglåser oppstår når..



En tråd kan sove ved å kalle på Thread...



Monitor er et nøkkelord i java



Tråder

- Lar oss parallellisere programmer
- Runnable spesifiserer hva som skal skje ved kjøring av tråden, `.run()`
- Starter en tråd ved å si `.start()`
- En tråd kan sove ved å kalle `.sleep()`, men dette kan kaste `InterruptedException`
- Vente på andre tråder: `.join()`

Kritiske regioner og delte ressurser

- Tråder kan dele data mellom seg, men dette kan skape rare situasjoner (race conditions)
- Bruker låser for å unngå uheldig oppførsel (ReentrantLocks)
- Tråden som har låst tråden, *må* låse opp tråden (unlock())

Monitorer, kondisjoner, barrierer

- Monitorer et designprinsipp som håndterer data forsvarlig
- Kondisjoner tilhører en lås, og har metodene: signal, signalAll, await
- Barrierer: CountdownLatch, evt. kalle .join()

Liveprogrammering

Kiosk med produsenter og konsumenter

Hva nå?

- Trix
- Gjør oblig 2 på nytt
- Tidligere eksamensoppgaver om tråder
- Oblig 6 vår 2019