

# Projects for Extreme Multiprogramming

Ideas and rules

# Exercises/Exam

- You will work on a project during the course – this will also be your exam project
- There will be no classic exercises
- The project must contain
  - Considerations on the concurrency available in the system
  - Design
  - Implementation
  - Findings

# Exercises/Exam

- Implementation can be in any language you'd like
- However there are languages that suit the model well
  - C++ - using CSP C++
  - Java – using JCSP or CTJ
  - Occam – the father of CSP languages

# Exercises/Exam

- You decide on the kind of project you want
  - But it need to be concurrent in nature...:D
- Groups are allowed (but don't tell the minister ;))

# Exercises/Exam

- Project examples
- Action game – 2D and event based is nice and easy:
  - Worm, ping-pong and similar oldies
  - **KEEP YOUR AMBITION LEVEL LOW!**
- Robotics
  - Library for handling sensor input from a robot
  - Plan to use a simulated environment

# Exercises/Exam

- Automata modeling
  - Conways life
  - WATOR
- Networking
  - Webserver, IP-phone central, etc

# Processing engines

- Farmer harvester model
- Implement a system for producer consumer problems
- Computationally bound
- Service bound
- Alpha-beta searches

# Rules

- Your project have to apply the CSP model taught in this class
- This means
  - No shared datastructures
  - Only channel communication
  - Considerations towards reuse of components
  - Hopefully open-ended parallelism

# Rules

- KISS
  - Maximum pages in report:
    - 1 student 10 pages
    - 2 students 15 pages
    - 3 students 20 pages