Concurrent and Distributed Systems
Course Outline - Student Version

Kenneth Skovhede
Brian Vinter
David Marchant
Niels Bohr Institutet

2018/2019
Block: 2
Schedule Group: B

Document Updated: 2019-01-08 22:42:26.590539

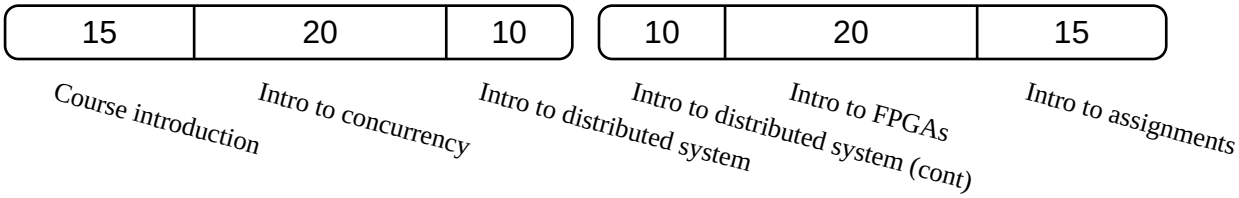
Lecture 1: Tuesday 20 November, Week 1 (47)

Location: Aud B, Blegdamsvej
Lecturers: Kenneth Skovhede, Brian Vinter, and David Marchant

Outline

- General introduction to the course
- Definition of concurrency and overview of issues and potential solutions
- Introduction to challenges and methods with distributed systems
- Brief description of FPGAs and what problems they can solve
- Introduction to the assignments
- Introduction to the hardware platform

Lecture Plan



Practical 1: Tuesday 20 November, Week 1 (47)

Location: Aud C, Blegdamsvej
Supervisors: Kenneth Skovhede, Brian Vinter, David Marchant, and Carl-Johannes Johnsen

Notes

Getting started

- Help with connection to Jupyter
- Help with PyCSP
- Using the Pynq
- Accessing the microphones

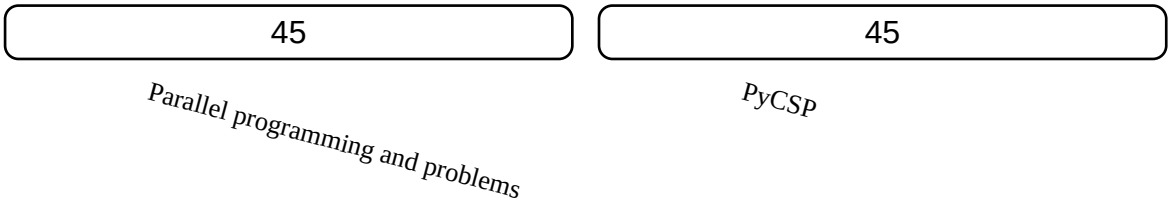
Lecture 2: Friday 23 November, Week 1 (47)

Location: Aud B, Blegdamsvej
Lecturers: David Marchant

Outline

- Parallel programming and problems
- PyCSP

Lecture Plan



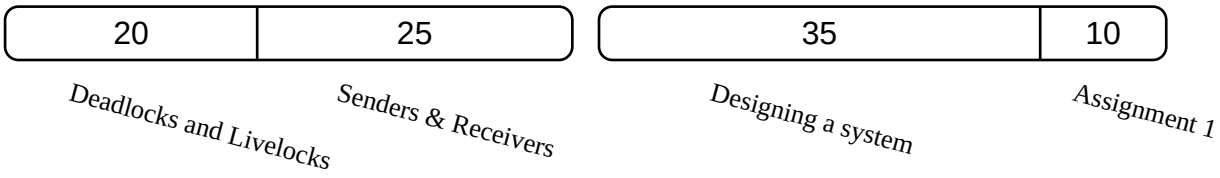
Lecture 3: Tuesday 27 November, Week 2 (48)

Location: Aud B, Blegdamsvej
Lecturers: David Marchant

Outline

- Deadlock
- Livelock
- Senders & Receivers
- Designing a system

Lecture Plan



Practical 2: Tuesday 27 November, Week 2 (48)

Location: Aud C, Blegdamsvej
Supervisors: Kenneth Skovhede, Brian Vinter, David Marchant, and Carl-Johannes Johnsen

Notes

Getting started

- Help with connection to Jupyter
- Help with PyCSP
- Using the Pynq
- Accessing the microphones

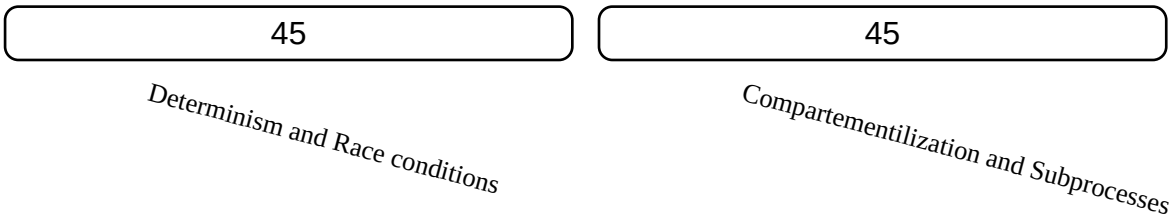
Lecture 4: Friday 30 November, Week 2 (48)

Location: Aud B, Blegdamsvej
Lecturers: David Marchant

Outline

- Determinism
- Race conditions
- Compartementilization
- Subprocesses

Lecture Plan



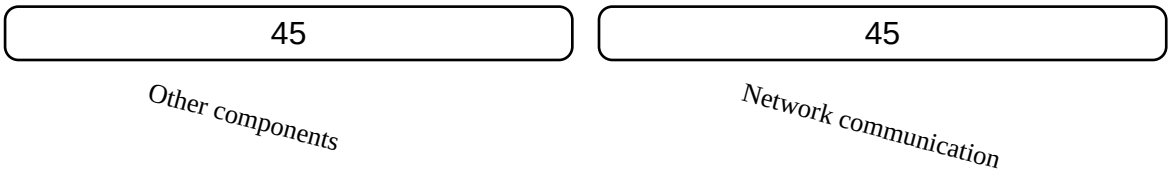
Lecture 5: Tuesday 4 December, Week 3 (49)

Location: Aud B, Blegdamsvej
Lecturers: David Marchant

Outline

- Barriers
- Agents
- Preconditions
- Network communication

Lecture Plan



Practical 3: Tuesday 4 December, Week 3 (49)

Location: Aud C, Blegdamsvej
Supervisors: Kenneth Skovhede, Brian Vinter, David Marchant, and Carl-Johannes Johnsen

Notes

Getting started

- Help with connection to Jupyter
- Using the Pynq
- Computer Networks

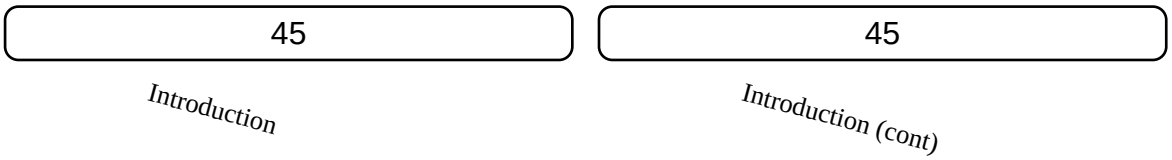
Lecture 6: Friday 7 December, Week 3 (49)

Location: Aud B, Blegdamsvej
Lecturers: Brian Vinter

Outline

- Introduction
- Protocols
- Error models

Lecture Plan



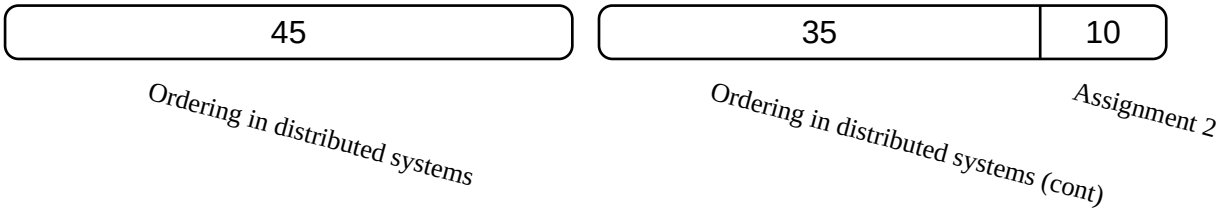
Lecture 7: Tuesday 11 December, Week 4 (50)

Location: Aud B, Blegdamsvej
Lecturers: Brian Vinter

Outline

- Causal and total ordering

Lecture Plan



Practical 4: Tuesday 11 December, Week 4 (50)

Location: Aud C, Blegdamsvej
Supervisors: Kenneth Skovhede, Brian Vinter, David Marchant, and Carl-Johannes Johnsen

Notes

Getting started

- Help with connection to Jupyter
- Using the Pynq
- Computer Networks

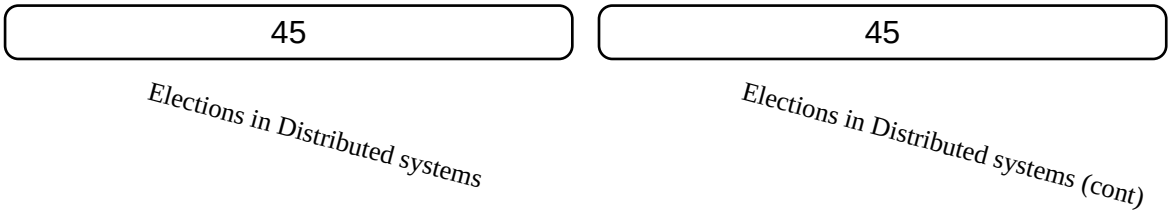
Lecture 8: Friday 14 December, Week 4 (50)

Location: Aud B, Blegdamsvej
Lecturers: Brian Vinter

Outline

- Elections
- Groups

Lecture Plan



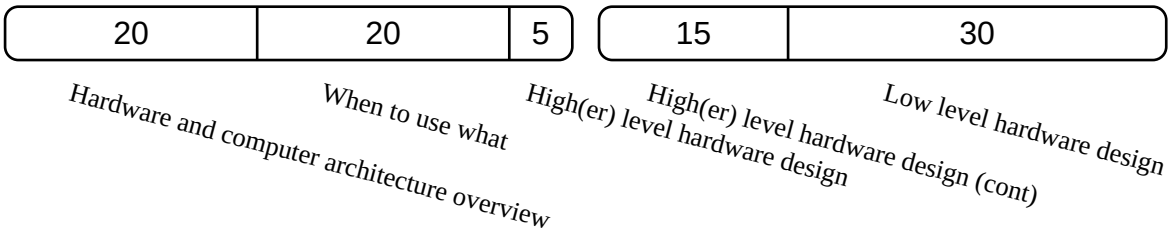
Lecture 9: Tuesday 18 December, Week 5 (51)

Location: Aud B, Blegdamsvej
Lecturers: Kenneth Skovhede

Outline

- What hardware can we make? FPGA, ASIC, hybrid, etc.
- Brief computer architecture overview
- When is hardware better than CPU/GPGPU?
- When is hardware worse than CPU/GPGPU?
- From software to hardware
- Designing and implementing hardware
- SME and PyRTL
- Introduction to assignment 3

Lecture Plan



Practical 5: Tuesday 18 December, Week 5 (51)

Location: Aud C, Blegdamsvej
Supervisors: Kenneth Skovhede, Brian Vinter, David Marchant, and Carl-Johannes Johnsen

Notes

Getting started

- Help with connection to Jupyter
- Using the Pynq
- SME
- GHDL
- Vivado

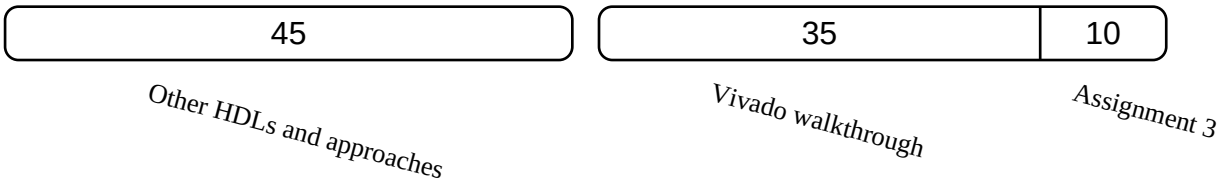
Lecture 10: Friday 21 December, Week 5 (51)

Location: Aud B, Blegdamsvej
Lecturers: Kenneth Skovhede

Outline

- VHDL and Verilog
- Designing hardware with HDLs
- Differences between HDLs
- Using abstractions for hardware
- SME and SMEIL
- Other approaches
- Using Vivado for testing, timing, mapping, integrating etc.

Lecture Plan



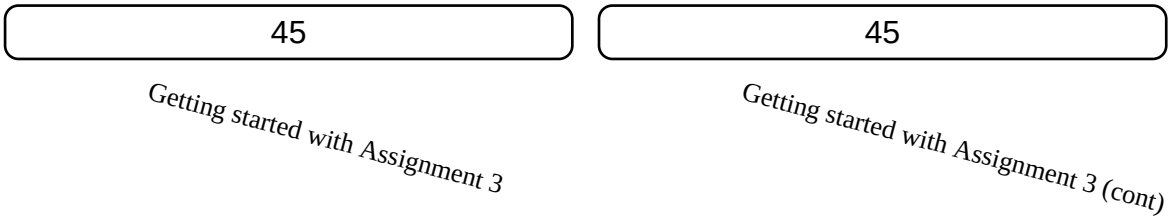
Lecture 11: Friday 4 January, Week 6 (1)

Location: Aud B, Blegdamsvej
Lecturers: Kenneth Skovhede

Outline

- Help with assignment 3

Lecture Plan



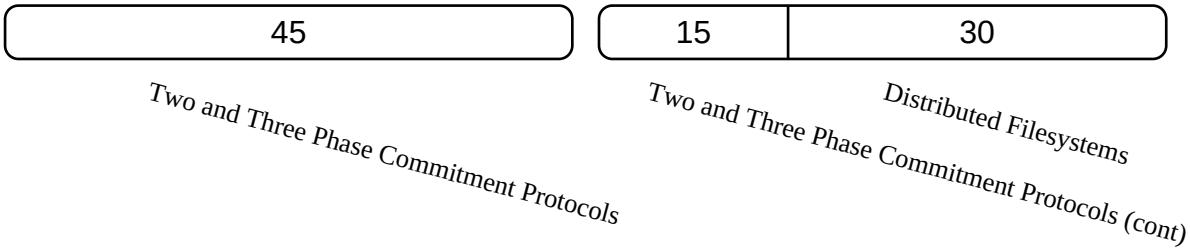
Lecture 12: Tuesday 8 January, Week 7 (2)

Location: Aud B, Blegdamsvej
Lecturers: Brian Vinter

Outline

- Two and Three Phase Commitment Protocols
- Distributed Filesystems

Lecture Plan



Practical 6: Tuesday 8 January, Week 7 (2)

Location: Aud C, Blegdamsvej
Supervisors: Kenneth Skovhede, Brian Vinter, David Marchant, and Carl-Johannes Johnsen

Notes

Getting started

- Help with connection to Jupyter
- Using the Pynq
- SME
- GHDL
- Vivado

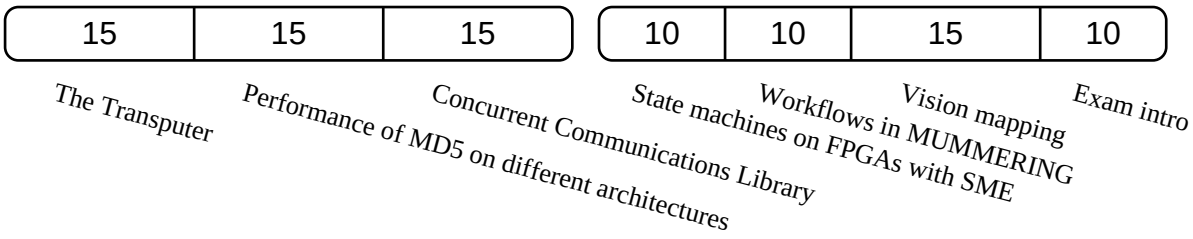
Lecture 13: Friday 11 January, Week 7 (2)

Location: Aud B, Blegdamsvej
Lecturers: Kenneth Skovhede, Brian Vinter, David Marchant, and Carl-Johannes Johnsen

Outline

- The Transputer, a processor for running CSP programs
- Comparing performance of MD5 hashing on CPU, GPGPU and FPGA with different programming models
- An introduction to CoCoL for writing scalable programs with a CSP inspired approach
- Building state machines in hardware using async/await statements
- Workflows for processing and analysis of Xray images in the MUMMERING project
- Designing a chip and processing system for handling input from multiple computer visision systems
- Introduction to the exam

Lecture Plan



Notes

In this lecture we present active research projects relevant for Concurrent and Distributed Computing. \nEach of the projects can be used as inspiration for student projects.\nAt the end we introduce the exam assignment.

Assignment 1 — The Cricket Locator - Part 1

Set: Tuesday 27 November
Due: Friday 7 December

Guesstimated Time Breakdown

1h	Familiarisation
4h	Finding and reading relevant RFCs and documentation
4h	Implementation and testing
4h	Writing report
Total: 13	hours

Learning Goals

- Hands-on experience with hardware
- Use PyCSP to write a program
- Using the Jupyter environment and communicating with a hardware device

Notes

In this first assignment you will be designing a 1-D cricket locator using a CSP design approach. \nYou will also be writing a report on your web client.

Assignment 2 — The Cricket Locator - Part 2

Set: Tuesday 11 December
Due: Friday 21 December

Guesstimated Time Breakdown

1h	Familiarisation
4h	Finding and reading relevant RFCs and documentation
5h	Implementation and testing
5h	Writing report
Total: 15	hours

Learning Goals

- Find, read, and understand material used for defining and describing Internet protocols, including RFCs
- Use and understand the socket library for a server
- Be able to implement a simple protocol directly using sockets
- Describe how a URL can map to a file system

Notes

In this second assignment you will be designing a 2-D cricket locator using a network of sensors. \nYou will also be writing a report on your webserver.

Assignment 3 — The Cricket Locator - Part 3

Set: Friday 4 January
Due: Friday 11 January

Guesstimated Time Breakdown

1h	Familiarisation
2h	Finding and reading relevant RFCs and documentation
8h	Implementation
5h	Writing report
Total: 16	hours

Learning Goals

- Working with clocked logic
- Understanding time evolution on signals

Notes

In this third assignment you will be designing a real version of the detector array you made in the first assignment, and implement it on actual hardware. \nWe will assume you have a setup similar to the distributed system in assignment two, and now need to develop the actual detector array.\nYou will also be writing a report on your results.

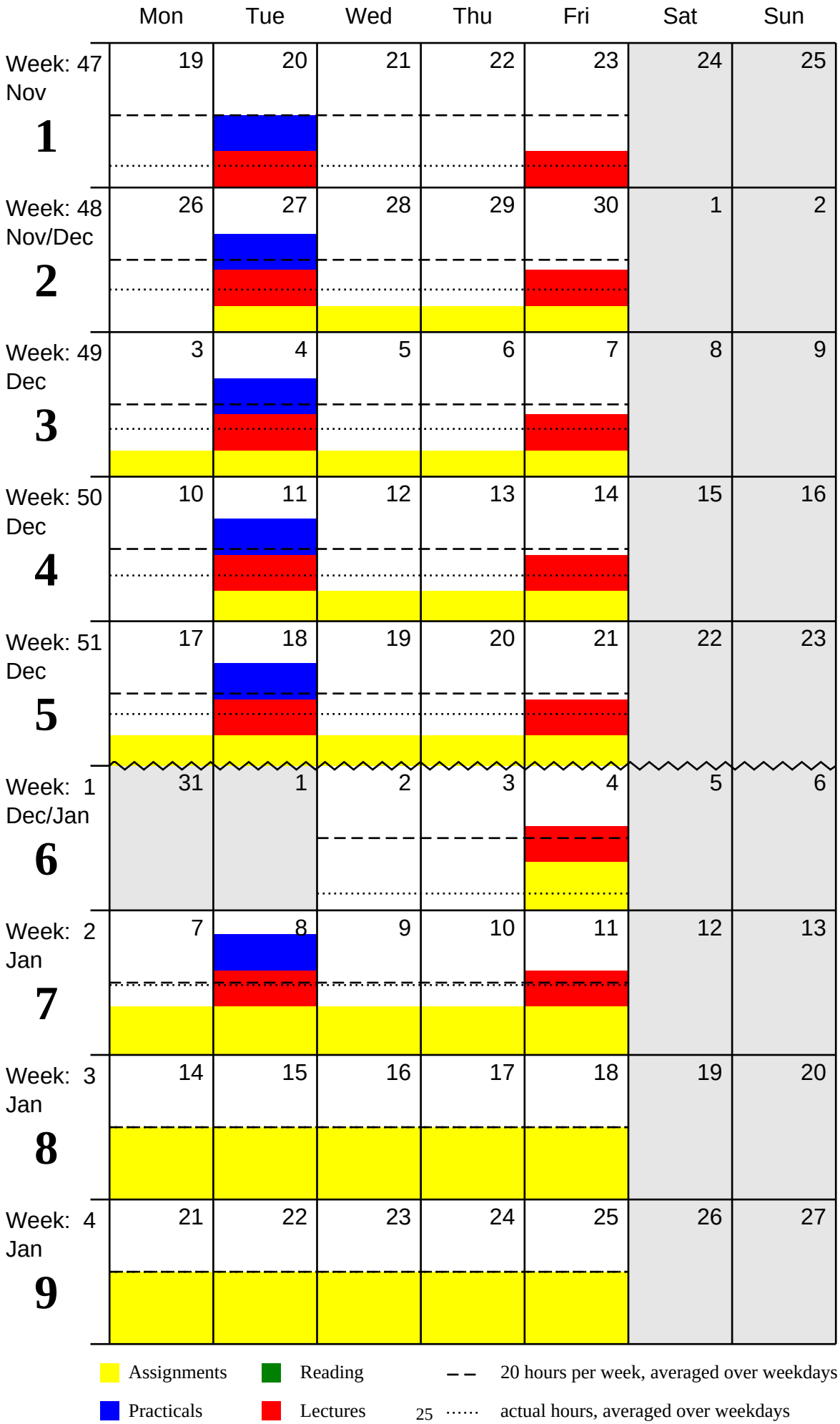
Exam

Set: Monday 14 January
Due: Friday 25 January

Guesstimated Time Breakdown

2h	Familiarisation
8h	Finding and reading relevant RFCs and documentation
25h	Implementation
5h	Writing report
Total: 40	hours

Guesstimated Work Pressure



Using this Document

We have made this document to help use (the lecturers) prepare the course. We have also, however, made it as a tool to communicate to you (the students) what we expect you to get out of the course, the individual lectures, practical session, and assignment. Thus, we have presented the overview of the content covered in each lecture and practical session, together with a number of learning goals. At the end of a lecture or practical session you can use these learning goals to take stock of what you have learnt and what you need to work on. Do not expect to have the learning goals covered by just going to the lectures without doing any auxiliary work, such as reading other material.

Updates

We will from time to time make updates to this document. You should check regularly on Absalon to see if you have the latest version. The version of the document can be seen on the front page of this PDF.

The Assignments

We have made some guesses as to how long the assignments as a whole should take and provided an estimate of how you may want to consider dividing your time amongst the individual parts of an assignment. These are *guesstimates* and should be taken as a *guide* only.

The Guesstimated Work Pressure

The guesstimated work pressure is divined by taking the time you are going to spend attending lectures and practical sessions, working on your assignments, and reading the set material in the book every week (with a generous amount of time allocated to read a page). Work on the assignments and required reading has been split evenly over the days from when it was set to when it is due. The assumption made in the diagram is that you work a five day week and do not work on the weekends. This may or may not reflect reality and you may of course work however and whenever you like.

You should note that the title of the diagram contains the word *guesstimate*, which a dictionary (*The New Oxford American Dictionary, Second Edition*) defines as: “an estimate based on a mixture of guesswork and calculation.” This is exactly what it is, an estimate of how much time we think you need to spend on the course, it is not a prescription! You may use more time or less, this is fine. Thus, the *Guesstimated Work Pressure* is there for you to be able to visualise the workload in the course and thereby help you arrange your work schedule. Please don’t throw a fit if my guesstimates turn out to be somewhat inaccurate, but do let us know so we can adjust them in future iterations of the course.

Providing Feedback

At the end of the course we will be asking for your feedback. We would like to know how useful this document was to you and how we could improve it. As already mentioned, we will be asking you how much time you spent on different parts of the course to see how they match up with our expectations as to how long I think they should take you. It would therefore be useful if you can keep (rough) notes on the time you spend, for example on reading and your assignments.