

Operating Systems

INF 3151, INF 4151

Administrative Introduction

<http://www.uio.no/studier/emner/matnat/ifi/INF3151/h06/>

My own OS

Who is helping you to learn

- Teachers:
 - Otto Anshus, otto@cs.uif.no
 - Vera Goebel, goebel@ifi.uio.no
 - Thomas Plagemann, plageman@ifi.uio.no
 - Guest lectures from Tore Larsen, Knut Omang, and John Markus Bjørndalen
- PhD students:
 - Matija Puzar, matija@ifi.uio.no
 - Ereğ Gökürk, erek@ifi.uio.no
 - Zeljko Vrba, zvrba@ifi.uio.no
- Teaching assistants (“gruppelærere”!)
 - Group : Asbjørn Sannes, asbjorsa@student.matnat.uio.no
 - Group : Jørn Uno Myrvoll, juno@student.matnat.uio.no
 - Group : Jia Zhiguang , zhiguang@student.matnat.uio.no



Learning by doing

- Guided process to build your OS
 - First design! You propose, we give you feedback!
 - Afterwards implementation
 - In total six projects
- Grading based on your presentation
 - Design (one week)
 - Code (two weeks, except P3 which has only 1 week)
 - Deliverables:
 - 1 printout in the box that is marked “innleveringer” under “luka”
 - 1 mail with all stuff to your TA
 - Deadline for hard copy: **Wednesday 12:00 local time SHARP!**



What means design?

- **Design, from Wikipedia, the free encyclopedia**
 - **Design**, usually considered in the context of the [applied arts](#), [engineering](#), [architecture](#), and other such [creative](#) endeavours, is used as both a [noun](#) and a [verb](#). "Design" as a verb refers to the process of originating and developing a plan for a new object (machine, building, product, etc.). As a noun, "design" is used both for the final plan or proposal (a drawing, model, or other description), or the result of implementing that plan or proposal (the object produced).
 - Designing normally requires considering [aesthetic](#), [functional](#), and many other aspects of an object, which usually requires considerable [research](#), [thought](#), [modeling](#), iterative adjustment, and re-design.



Design as process

- How to solve the assignments, i.e, develop a well working program?
- Think and discuss with your group mate
- Identify alternative approaches, e.g, important data structures, algorithms, etc.
- Evaluate the approaches and select the best
- Document the main results of this process in your design proposal



Design proposal

- Mandatory deliverable for each project
- Not more than 10 pages!
- Description of your plan of how to solve the problem and why in this way
- Typically this document contains:
 - Brief description of the different alternatives you have studied and why you selected which.
 - Detailed textual description of the proposed data structures and algorithm(s) to be used with supporting illustrations in form of figures, flow diagrams, or pseudo code. If you use standard data structures or algorithms, put your main emphasis on how they are applied to the problem.
 - Description of what functionality will be implemented in which file/function, and how these implemented parts will interact (will they work exclusively? Concurrent? Can they be interrupted? etc.) to attain the goal given in the problem description.
 - Key details such as why a particular mask value is chosen, how it is constructed, how and why a particular register is loaded with a particular value, etc.
- Often you will also get some hints during the presentation of the project on what should be addressed in the design proposal

What do we do with the design proposal?

- Give you early feedback
 - Oral presentation of your design proposal to your TA
 - TA gives you feedback whether you are on the right track or not
 - saves you a lot of time
 - helps you to get a better grade
- Give you a partial grade
 - In all projects the code counts 50% more for the grade than the design proposal

Grading of Exercises

- All TAs will give the same amount of support for students
Read: we help you to learn, but not to make shortcuts!
- (Additional help for “desperate” students might be reflected in the grade)
- Each deliverable is graded by a PhD student
- External censor controls randomly
- At the end of the term all grades are combined and eventually adapted

Group Lectures

- Each group should have ideally not more than 20 students
- Right now we follow the airline approach: overbooking
- Deliverables have to be prepared by teams of two students
- Teams of three students are not allowed because of grading

Exception Handling - I

- Sick leave:
 - Official certificate from a medical doctor
 - Oral examination about the missed deliverable
- Disagreement in a group:
 - Oral examination
- Cheating / fraud:
 - According to rules of Faculty of Mathematical and Natural Sciences
 - The declaration you sign is just to make you aware of the existing rules



The Big NoNo



- It is not allowed to distributed code from the assignments or to make it accessible to others (except lecturers and teaching assistants of the course) neither in paper form nor in electronic form. This is valid for the code developed from the students as well as code distributed as part of the assignments.
- All contraventions are regarded as fraud!
- You have to sign a corresponding declaration and deliver it with P1 theory assignment to your TA.

The Big YesYes

- Start to work hard right from the beginning
- Be active in the group lectures
- Discuss with your partner
- Discuss with other students (but do not exchange code or the answers to the theory assignments!)
- Solving problems and understanding an OS can be a lot of fun!

Competition

- Date: 14/12
- Who has developed the fastest kernel?
 - Running benchmarks to test
 - Benchmark will be available for tuning
- Telenor Research and Development sponsors with 5 coupons from Telehuset, each for kr. 2000,-



Hall of OS Fame



Questions?

- Take a look at the FAQ:
 - <http://www.ifi.uio.no/~inf3151/afaq.html>

