Joint Master's Programme in Software Engineering
Overview of courses – 2016-08-19

- Supervisors: HiB or UiB or joint, possibly also external
- Two branches, determins compulsory courses
  - PU – software engineering:
    INF234(A), MOD250(A), MOD251(S)
  - PUT – programming theory: INF234(A) and 3 of
    INF220(A), INF225(I-A1), INF329(I-A1),
    INF223(S), INF227(S), or INF210(I)
- Remaining courses freely selected from HiB / UiB
- Two master thesis variants (talk to supervisor)
  - Long: 60 stp, deadline 1 June 2018
    Research/development oriented, needed for PhD
  - Short: 30 stp, strict start/end dates
    More structured studies, 3 extra courses

Courses – 1st semester

Autumn 2016

– • INF 220 Program specification (Magne Haveraaen)
– • INF 225 Program Translation (Anya Bagge)
– INF 226 Software security (Samson Gejibo)
– • INF 329 Selected Topics in Programming Theory (Jaakko Järvi)
– *•INF 234 Algorithms
– * MOD 250 Advanced Software Technologies
– MOD 252 Agent Technologies
– MOD 351 Introduction to Grid and Cloud Computing

All semesters (find supervisor)

– INF 219 Programming Project (possibly bachelor)
– INF 319 Programming Project (master)

Courses – 2nd semester

Spring 2017
- INF 222 Programming Languages (Jaakko Järvi) = INF 329 (A1)
- • INF 223 Category Theory (Uwe Wolter)
- • INF 227 Introduction to logic (Marc Bezem?)
- *MOD 251 Modern Software Development Methods
- MOD 350 Model driven Software Development

All semesters (find supervisor)
- INF 219 Programming Project (possibly bachelor)
- INF 319 Programming Project (master)

Courses – irregular

Irregular semesters (ask supervisor)
– INF 210 Modelling of Computing
– INF 328 Elements of Programming Languages
– INF 329 Selected Topics in Programming Theory

Some fun/filler courses
– INF/INFO 207 Social Networks Theory (autumn)
– INF 236 Parallell programmering (spring) – requires INF 234
– INF 250 Foundations of data-oriented visual computing (spring)
– INF 251 Computer Graphics (autumn) - requires INF 250
– INF 283 Introduction to Machine Learning (autumn)

INF 220 Program specification – 1st/3rd semester

The course introduces methods for software specification, with a focus on abstraction in the software process. Covers the early stages of software development: domain enginering, (formal) requirements specifications, and high integrity applications. You will get an in depth understanding of algebraic specifications, generic programming and correctness.

The course uses the research language Magnolia – central to several thesis topics.

Autumn 2016: Magne Haveraaen
First lecture: Tuesday 23 August at 1415-1600, room 209M1

INF 225 Program Translation – only 1st semester

Vi ser på hvordan man lager programvare som kan prosessere programvare; slik som f.eks. kompilatorer, tolker, analyse- eller refaktoreringsverktøy. I år ser vi litt ekstra på analyseteknikker, til bruk f.eks. for å forstå programvare, eller for å finne bugs eller sikkerhetsfeil.

Kurset er praktisk rettet med hands-on oppgaver.

Autumn 2016: Anya Bagge
First lecture: Monday 22 August at 1015-1200, room 209N1

INF 226 Software security – 1st/3rd semester

This course we will explore the foundations of software security. We will consider important software vulnerabilities and attacks that exploit them – such as cross-site request forgery, buffer overflows, SQL injection, session hijacking, and so on. We will consider defenses that prevent or mitigate these type of attacks, including advanced testing and program analysis techniques. Importantly, we take a "build security in" mentality, considering techniques at each phase of the development cycle that can be used to strengthen the security of software systems.

Autumn 2016: Samson Gejibo
First lecture: Thursday 25 August at 1215-1600, room 209N1

INF 329/222 Programming Languages – 1st/2nd semester

The course gives a practical and theoretical introduction to the semantics of programming language constructs. The course covers a range from micro level constructs, e.g., parameter passing, to macro level constructs, e.g., module systems. Reflecting on the influence of the constructs on software quality forms part of the course. We discuss the notions of different approaches to types and type checking in languages. Some features we study by using them in programs, some by implementing them in small interpreters in Haskell. Occasionally we will work on other languages as well.

The course will help you to become more productive with the languages of today and more prepared to learn (perhaps also implement) new programming languages in the future.

Autumn 2016/spring 2017: Jaakko Järvi
First lecture: Tuesday 23 August at 1015-1200, room 209M1

INF 219/319 Project in informatics – 1st/2nd/3rd semester

Individual students or small teams solve a small problem, such as:

– Student initiated project
– Project proposed by external company / organisation
– A small part of a possible master thesis topic
– Supervisor initiated project, examples:

- Anya: "Play" with a sphero robot.
- Magne: "Play" with an NVIDIA Jetson TX1.
- Jaakko: Apply HotDrink property models (a JavaSscript library) to implement non-trivial user interfaces.
  Property models are a new approach to user interface programming. In this approach, the programmer writes a declarative specification of the user interface's data and its dependencies, instead of a complex network of event handlers.

Any semester: find a supervisor at UiB

No lectures

INF 223 Category Theory – 2nd semester

The course covers

- Diagrams, i.e., graph structures, are omnipresent in SE and programming.
- As models they play a central role in MDE.
- Compositional directed relations between models as well as model transformations constitute the kernel of MDE.
- The course introduces a mathematical language and theory to describe and reason about models, their relationships and transformations.
- This language is based on the concepts "directed graph" and "category", i.e., directed graphs with identity arrows and a composition of arrows.

Spring 2017: Uwe Wolter
First lecture: next semester

INF 227 Introduction to Logic – 2nd semester

The course gives a thorough introduction in the basic logics that are important for computer science:

- Propositional logic, which is underlying all digital hardware;
- Predicate logic, which is the logic of, for example, loop invariants and pre/postconditions (assertions in Java). Moreover, predicate logic is used in relational databases and many query languages.

The course provides the basis for understanding the many other links between logic and computer science.

For example, the propositional satisfiability problem formed the start of algorithmic complexity. The unsolvability of the halting problem is tightly connected to the undecidability of predicate logic. Temporal logics allow reasoning about time: linear time for sequential programs and branching time for concurrent programs. Description logics underpin the ontologies used in the semantic web.

Spring 2017: TBD
First lecture: next semester

Next week: first lectures

- • INF220 Tuesday 23 August at 1415-1600, room 209M1
- • INF225 Monday 22 August at 1015-1200, room 209N1
- INF226 Thursday 25 August at 1215-1600, room 209N1
- •*INF234 Monday 22 August 1215-1400, room "Stort auditorium"
- • INF329 Tuesday 23 August at 1015-1200, room 209M1
- * MOD250 Tuesday 23 August at 1015, room E206
- MOD252 Tuesday 23 August at 0815, room J202
- MOD351 Thursday 25 August at 0815, room J202
- INF219/319: find a supervisor at UiB

http://put.ii.uib.no/