

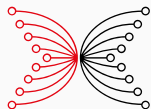
Welcome

Haskell and Cryptocurrencies

Dr. Andres Löb, Well-Typed LLP

Dr. Lars Brünjes, IOHK

2018-01-08



INPUT | OUTPUT

Goals

- Introductions (participants and instructors).
- Explain the course schedule.
- Clarify expectations and workload.

Introductions

Schedule

This week: kick-off week

- Full schedule: lectures in mornings and afternoons.
- Lab time every day.
- Small exercises, reviewed immediately.

Schedule today

- Introduction to Cryptocurrencies.
- Overview of Haskell.
- Lunch break.
- Lab setup (get everything installed, first steps).
- First set of small assignments.

Schedule is preliminary, and will be refined further:

- Monday, Wednesday, Friday morning: lectures.
- Tuesday morning: discuss assignments.
- Thursday morning: student status reports.
- Afternoons: Additional discussions and team work time.

There will be two coffee breaks each day at 11am and at 3pm.

- Prof. Philip Wadler, February 5th-9th
 - Propositions as Types
 - Category Theory for the Working Hacker
- Darryl McAdams, February 19th-23rd
 - Equational Reasoning
 - DSL's

Expectations and assignments

Mutual expectations

This is a full-time course.

- Work on assignments and projects is expected and necessary.
- Ask a lot of questions.
- Feel free to suggest ideas for course content.
- Course setup is somewhat flexible.
- Get proper feedback on your performance.

Weekly assignments

- Every Friday, one set of assignments will be distributed.
- To be handed in (electronically) on Friday one week later.
- To be solved individually.
- Reviewed by us.

Project assignments

- In larger intervals, larger project tasks will be distributed.
- To be worked on in teams (probably two teams in total).
- Teams will be determined at the end of the first week.
- Milestones will be agreed on.
- Use the afternoons to coordinate the team work.
- Intermediate and final demos / presentations here in the course.

Extra and bonus work

We may do additional things such as:

- Give everyone (or a few people) a paper to read and then discuss it in the course.
- Pose additional programming challenges and let participants present their solutions.
- Have more guest lectures, or watch a classic talk video together.
- ...

- Slack (questions, discussions)
- Github/ Google Drive (distribution of materials)