Course introduction

Functional Programming 2019-2020

```
import Data.Char(toUpper)

mkWelcomeMessage stylize year currentN lastN =
    concat
      [ stylize "Welcome"
      , " to INFOFP in " ++ show year ++ "!\n\n"
      , "We have " ++ show delta
      , " more students than last year!"
      ]
  where delta = currentN - lastN

welcomeMessage =
    mkWelcomeMessage capitalize 2020 348 291
  where capitalize = map toUpper
```

WELCOME to INFOFP in 2020!

We have 57 more students than last year!

The team

Matthijs Vakar and Frank Staals (me) in the lectures

- Contact us through email
- We both speak Dutch

10 teaching assistants in the labs

- Most of them are Dutch speakers

Guest lecture at the end of the course

Schedule

Lectures: twice per week

- Tuesday, 9.00 to 10.45
- Thursday, 13.15 to 15.00
- 15-minute break in the middle

Practicals and Instructions: twice per week

- Tuesday, 11.00 to 12.45 (online)
- Thursday, 17.15 to 19.00 (at USP)

You are expected to work at home/library/café/...

Communication channels

http://www.cs.uu.nl/docs/vakken/fp

- All important information is found there
- Schedule, slides, assignments, exercises

E-mail for important news

- Check your UU-mail regularly

Teams

- Practicals and Instructions through Teams

Resources

1. **Slides** contain most of the content
   - In some cases, supplemented by additional material
2. Pen-and-paper **exercises**
   - There's more than programming in this course
   - Ask questions during werkcollege sessions
   - Remember: there is no compiler at the exam
3. Book: Programming in Haskell (2nd edition) by Graham Hutton
   - The course follows it, except for chapters 13 and 17
   - More resources can be found in the website

Practical assignments

1. The first one helps you getting started
2. Three small ones with DOMJudge, one per week
3. One bigger project at the end

DOMJudge assignments

- Submissions are **individual**
  - Do not plagiarize!
- Graded mostly automatically, almost instant output
  - P1, P3: correct = 10, not fully correct = 1
  - P2, correct = 6, remaining 4 points for style

Style

- Hints in DOMJudge Automatic checks for good style
- Ask TAs for advice during practicals
- Important part of the final project grade

Final project

Develop your own **game** in Haskell

- Work in **pairs** is allowed and recommended
- 80% of your grade for practicals
- Submission in two parts
  1. Preliminary design document
  2. Code of the project

Tools

- Haskell as a programming environment
  - We use GHC, the de facto standard compiler
  - More information later
- HLint to check style
- Two different systems for submission
  - DOMJudge for automatic grading
  - Blackboard for final project

Rooms for instructions

- Double-check you assigned group in Blackboard!

Optional assignment

Learn and explain a Haskell library or language feature

- Up to additional 0.5 points for the final grade
- Work in groups of at most three
- More details after mid-term exam

Grading

Linear combination of three grades

- Theory T = 0.3 × midterm + 0.7 × final
- Practical P = 0.2 × DOMJudge + 0.8 × final
- Optional assignment O

Final grade F = 0.5 × T + 0.5 × P + 0.05 × O

To pass the course, you need

- F >= 5.5, T >= 5, P >= 5
- Pass at least two DOMJudge assignments

All other cases are described in the website

If you did the course last year

- **Resubmit** your DOMJudge assignments
- Redo the **final project**
  - Using the same code as last year is not allowed
- Redo **all** the **exams**

Let’s get started!