Overview

Johan Montelius

KTH

VT24

Introduction

- Learning outcomes
- Literature
- Lectures
- Assignments
- More

Learning outcomes

After passing the course, students should be able to:

- use recursion, pattern matching and non-modifiable data structures upon implementation in a functional programming language.

For higher grades, the student should also be able to

- use functions as first order objects and work with the functions of higher order
- explain the basics of functional programming, its structure and operational semantics
- evaluate functions with regard to time complexity
- use message based multi-threaded programming.

why functional programming

Why do we need a course in functional and concurrent programming?

why concurrency

Two reasons:

A tool to model interactive services.

Hardware can utilize concurrency to speed-up computations.

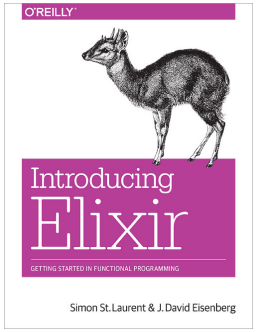
To build a good game engine, you need to master concurrency

Literature

This course is not about a particular language but ....

... we need a language to experiment with.

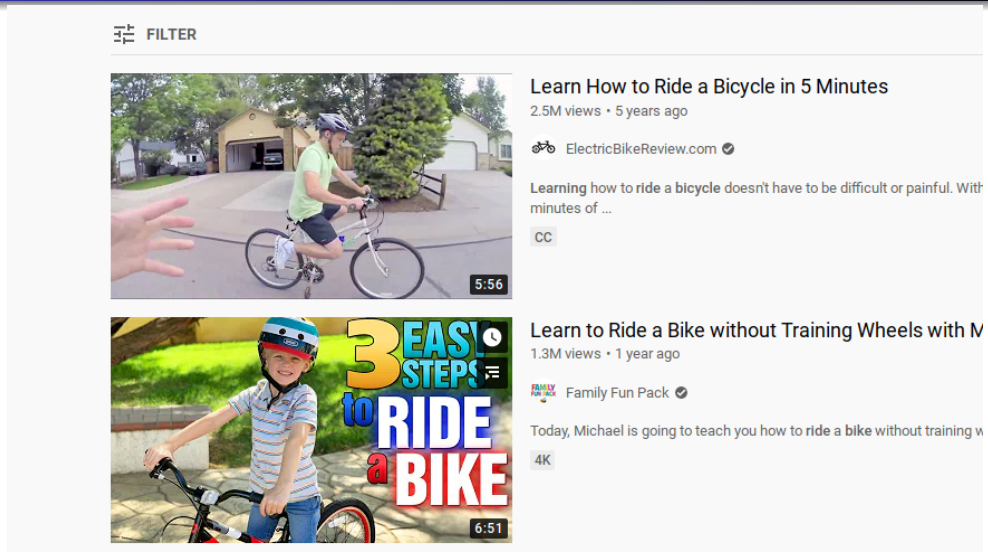
- Introducing Elixir - Getting Started in Functional Programming
- Simon St. Laurent, J. Eisenberg
- O'Reilly Media



Course material

- The web: elixir-lang.org
- Canvas: all question in the discussion board

learn how to bike



Lectures

We will lectures that will cover the following aspects:

- Functional programming
- Operational semantics
- Higher order programming
- Complexity
- Concurrency
- Parallelism

... most lectures available as videos.

Assignments to pass the course

A set of hand in assignments.

A short, 2-4 pages (4 for higher grade), report written in LaTeX following a template.

No, not four pages of code!

Explain what you did, why and what you learned.

There is a quizz that should be don by Friday.

Assignments - deadline every Friday

To pass the course:

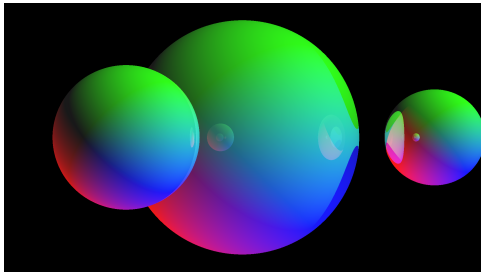
- Derivative - representing data, recursion.
- A key-value store
- Evaluate an expression.
- Operations on list - part I
- Advent of Code : Springs - part I
- Advent of Code : Ranges - part -I (II)
- Mandelbrot
- Huffman coding
- ...???

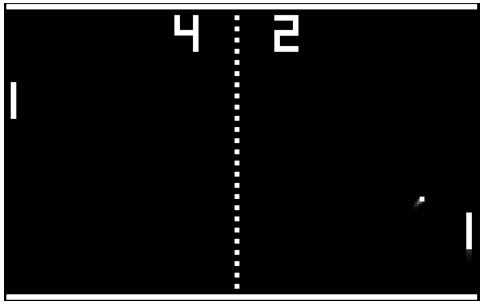
For higher grades:

- Meta interpreter
- Higher order (Operations on lists - II)
- Dynamic programming (Springs - part I)
- Dining philosophers

The seminars are not compulsory handing in the report is.

more seminars





course board

Two student representatives for course board. We will meet a couple of times during the course so that you can give feedback.

finally

Start programing today.