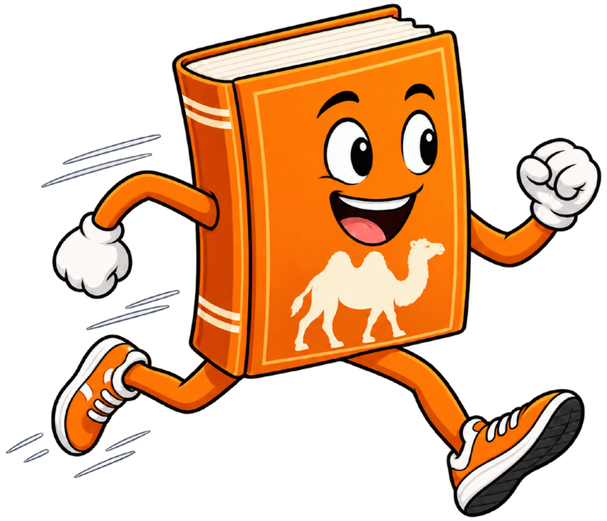
Textbooks that run

Interactive 0-install CS education in the browser



KC Sivaramakrishnan
26th September 2026
IndiaFOSS 6.0, Bengaluru

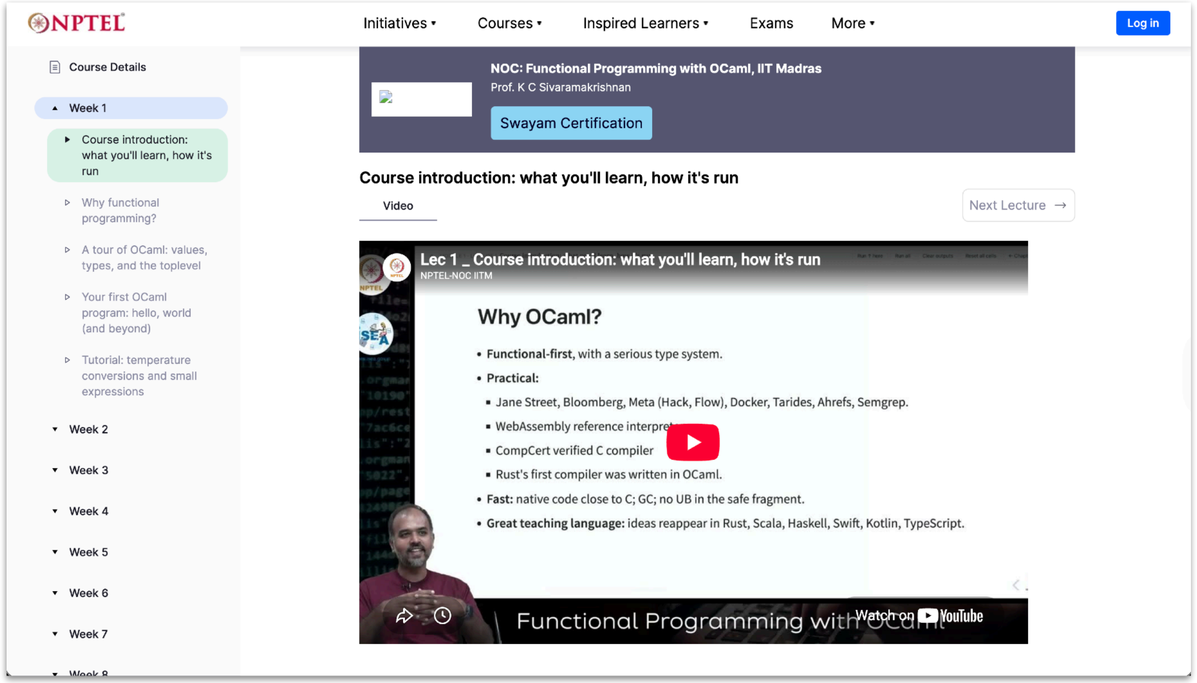
FP Launchpad

IIT MADRAS

Tarides

Story of the book

Functional Programming with OCaml

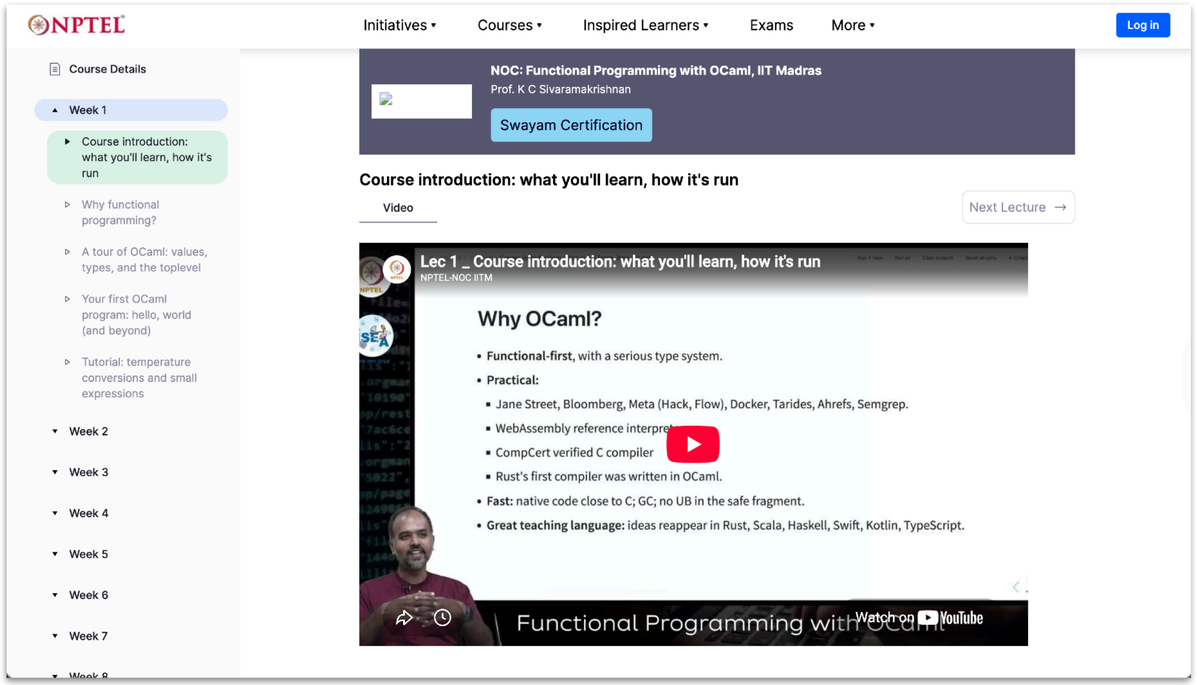


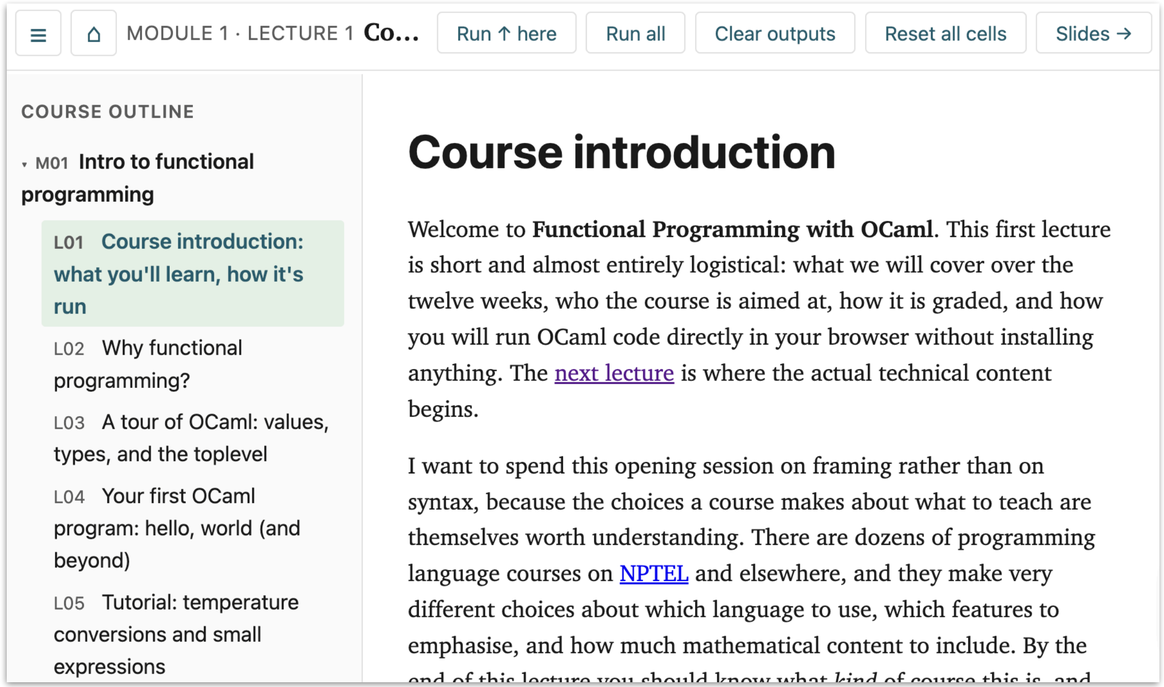
https://nptel.ac.in/courses/106106002

1227 Enrolled

Story of the book

Functional Programming with OCaml





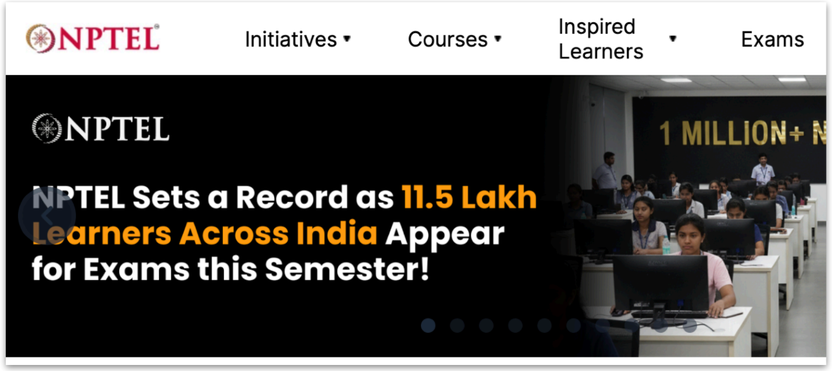
https://nptel.ac.in/courses/106106002

https://fplaunchpad.org/ocaml_nptel

1227 Enrolled

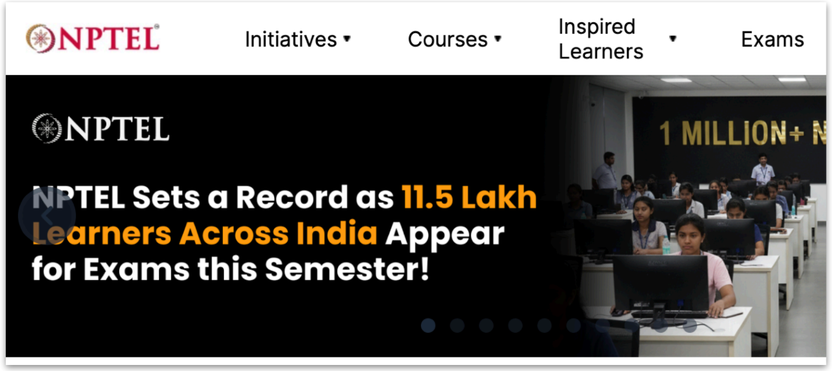
NPTEL

- A free online MOOC platform with courses from IITs and IISc
  - Videos are available on YouTube
  - A student can be “certified” by taking a proctored exam ₹1k per course
  - Credits can satisfy degrees in other colleges and universities



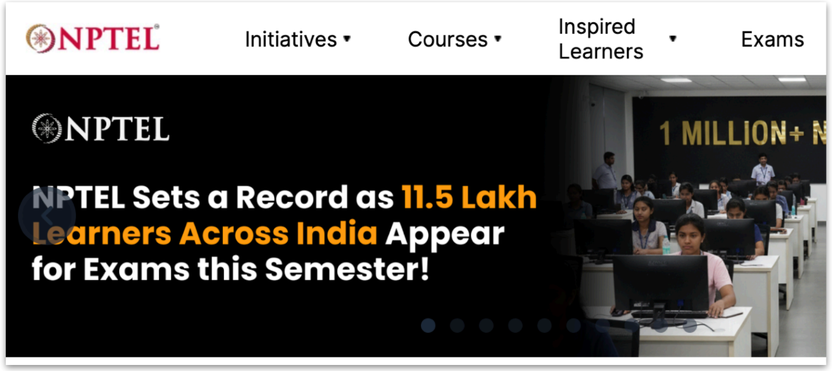
NPTEL

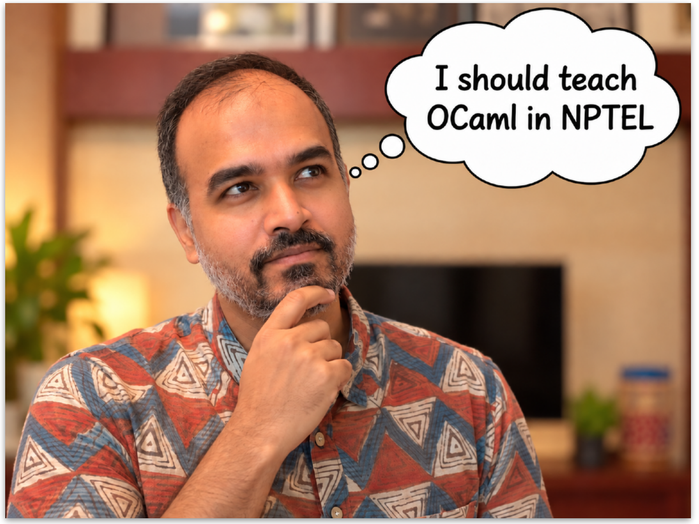
- A free online MOOC platform with courses from IITs and IISc
  - Videos are available on YouTube
  - A student can be “certified” by taking a proctored exam ₹1k per course
  - Credits can satisfy degrees in other colleges and universities
- **One of the greatest collective achievements of Indian academia**



NPTEL

- A free online MOOC platform with courses from IITs and IISc
  - Videos are available on YouTube
  - A student can be “certified” by taking a proctored exam ₹1k per course
  - Credits can satisfy degrees in other colleges and universities
- **One of the greatest collective achievements of Indian academia**





Delivery challenge

- Programming languages are learnt by poking at it

Delivery challenge

- Programming languages are learnt by poking at it
- ***How to get OCaml platform to 1227 students?***
  - Shared computers or lack of it
  - platform diversity
  - lack of broader computer skills
  - unreliable Internet

Delivery challenge

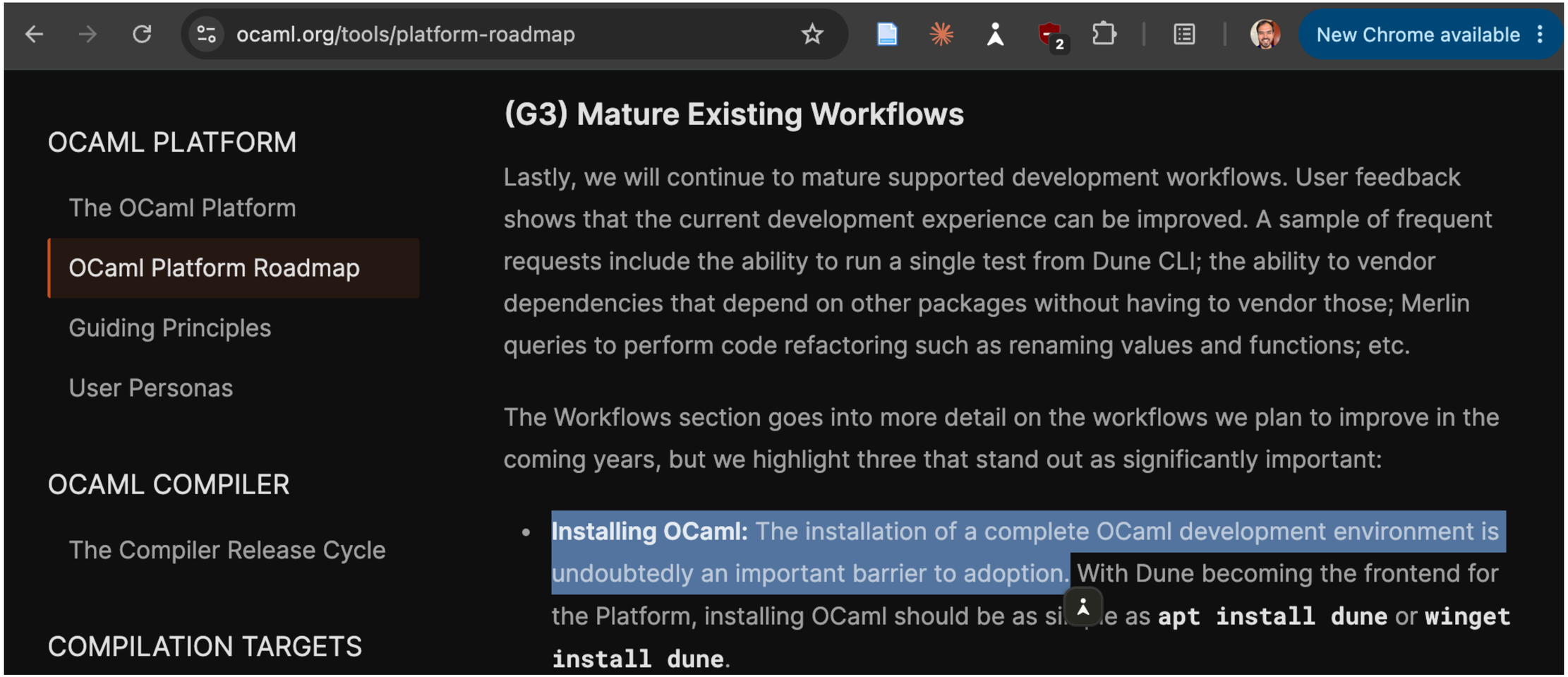
- Programming languages are learnt by poking at it
- ***How to get OCaml platform to 1227 students?***
  - Shared computers or lack of it
  - platform diversity
  - lack of broader computer skills
  - unreliable Internet
- And the millions (🤞) who will watch the videos later

Delivery challenge

- Programming languages are learnt by poking at it
- ***How to get OCaml platform to 1227 students?***
  - Shared computers or lack of it
  - platform diversity
  - lack of broader computer skills
  - unreliable Internet
- And the millions (🤞) who will watch the videos later
- Videos recorded in a studio
  - 2 TAs to run the course
  - Personalised attention is impossible

Installation challenge

- A big challenge for ***learning*** and ***adoption***

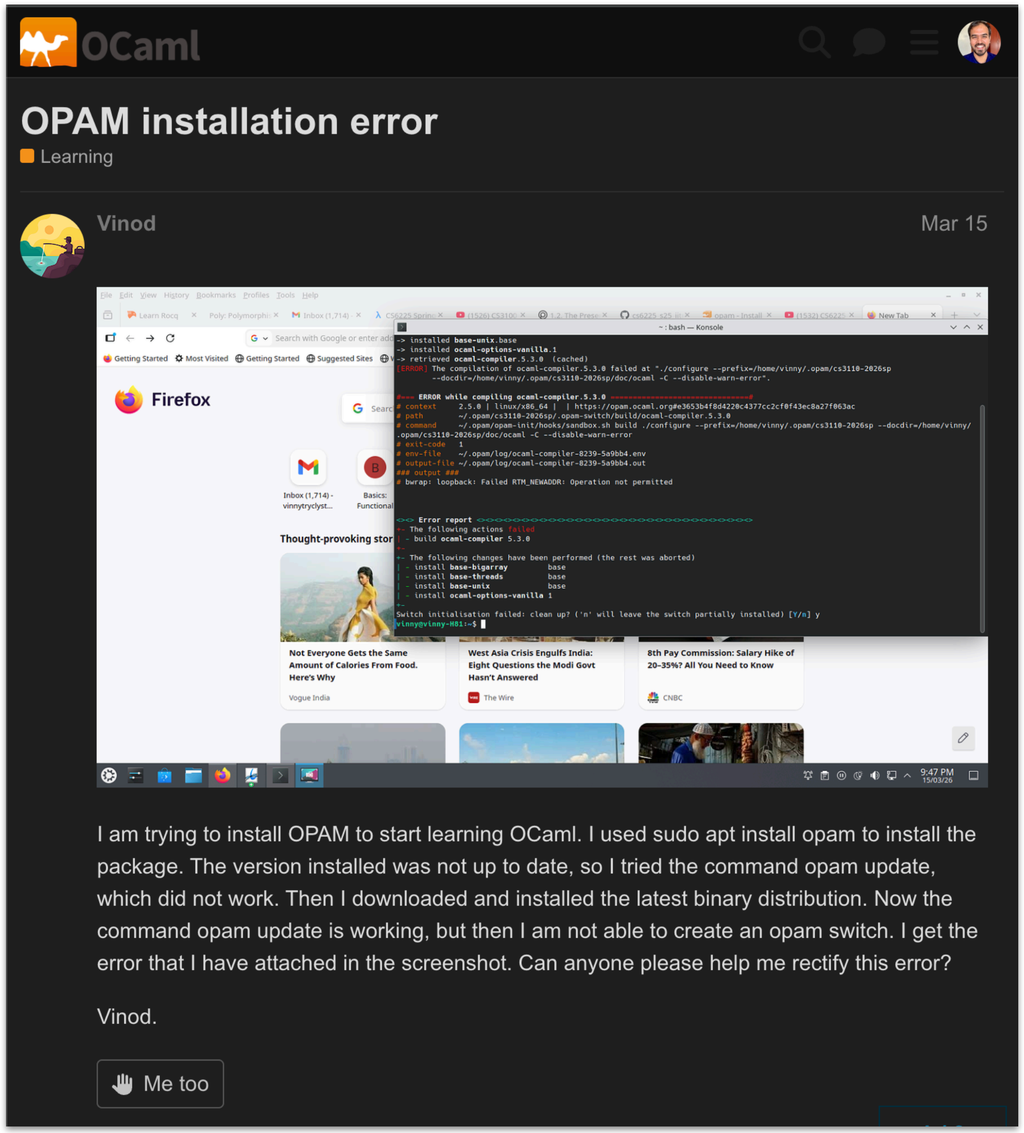


Installation challenge

- A big challenge for ***learning*** and ***adoption***
- Personal experience
  - IITM — many students unfamiliar with terminal
  - Hands-on OCaml workshops — 1/3rd of the time spent on installation

Installation challenge

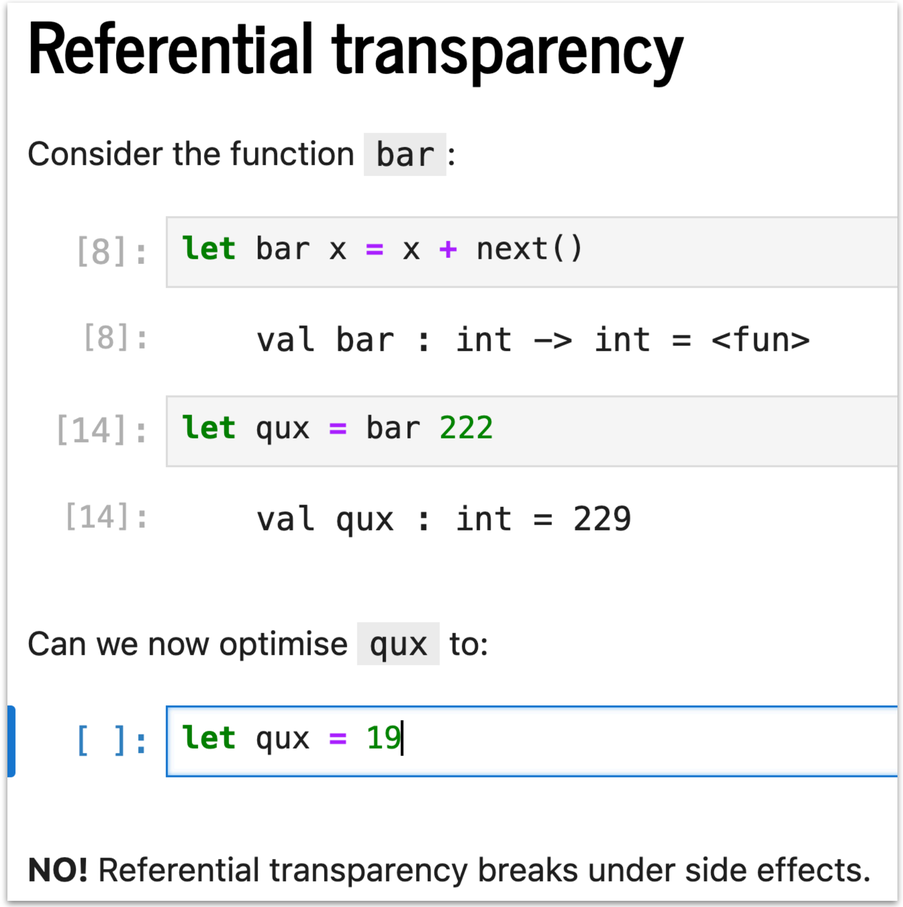
- A big challenge for ***learning*** and ***adoption***
- Personal experience
  - IITM — many students unfamiliar with terminal
  - Hands-on OCaml workshops — 1/3rd of the time spent on installation
- Users and students ***disengage*** when installation is a challenge
  - Especially challenging when students have little support



Who do we not hear from?

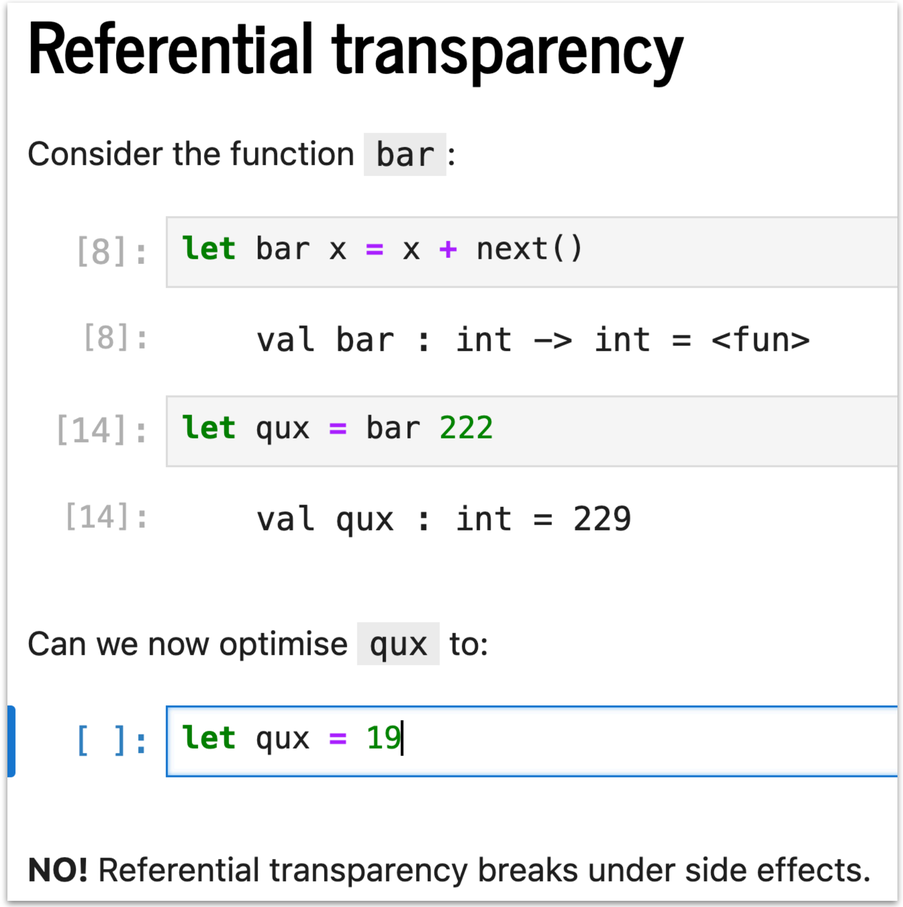
Earlier workflow

- CS3100 Paradigms of Programming
  - OCaml + Prolog
- Jupyter Notebooks + Docker
  - RISE for converting them to reveal.js slides
  - nbgrader for assignment evaluation
  - Several offerings at IITM with this set up.



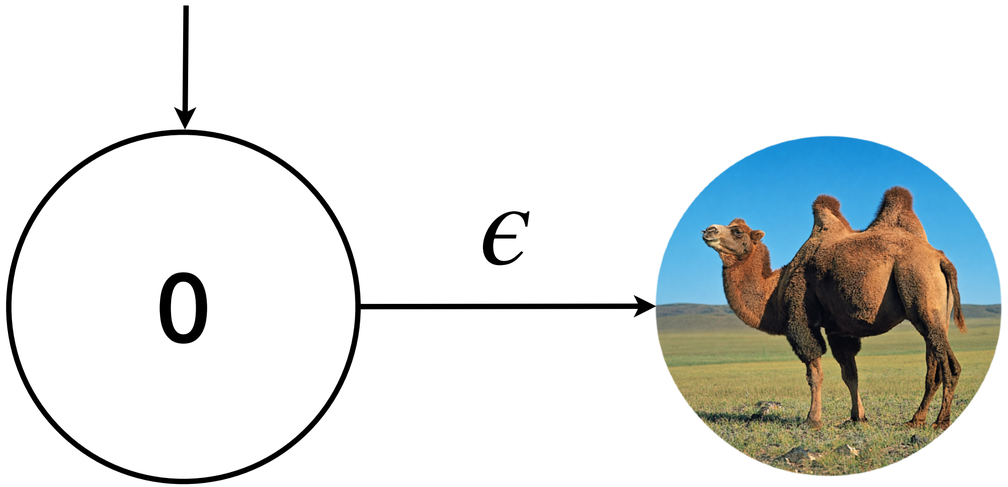
Earlier workflow

- CS3100 Paradigms of Programming
  - OCaml + Prolog
- Jupyter Notebooks + Docker
  - RISE for converting them to reveal.js slides
  - nbgrader for assignment evaluation
  - Several offerings at IITM with this set up.
- Downsides
  - Needs Docker — Image is ***2.86GB***
  - Editor is not language-aware — no types, live error, code completion
  - Not a book — cannot be read in isolation; meant to go along with lecture videos



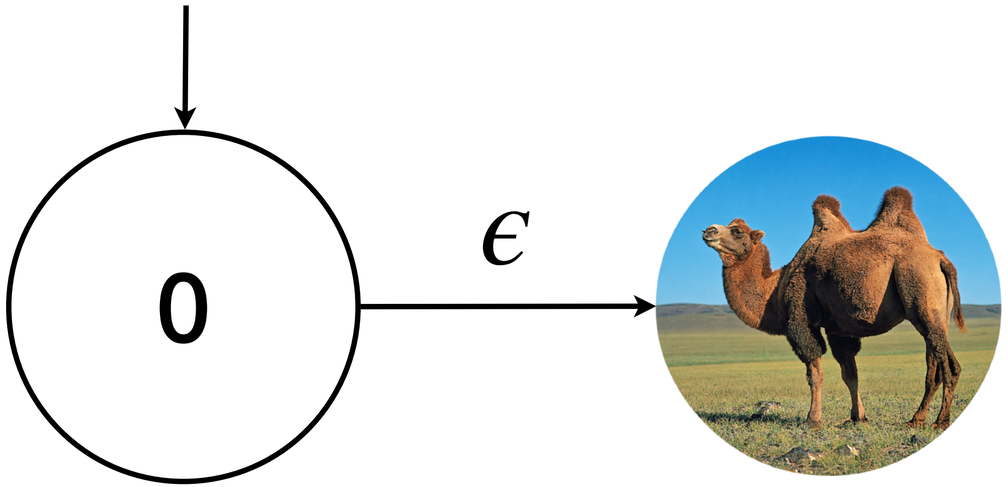
0 to OCaml in 0 steps

- **Start immediately**
  - Zero installation; runs offline in the browser



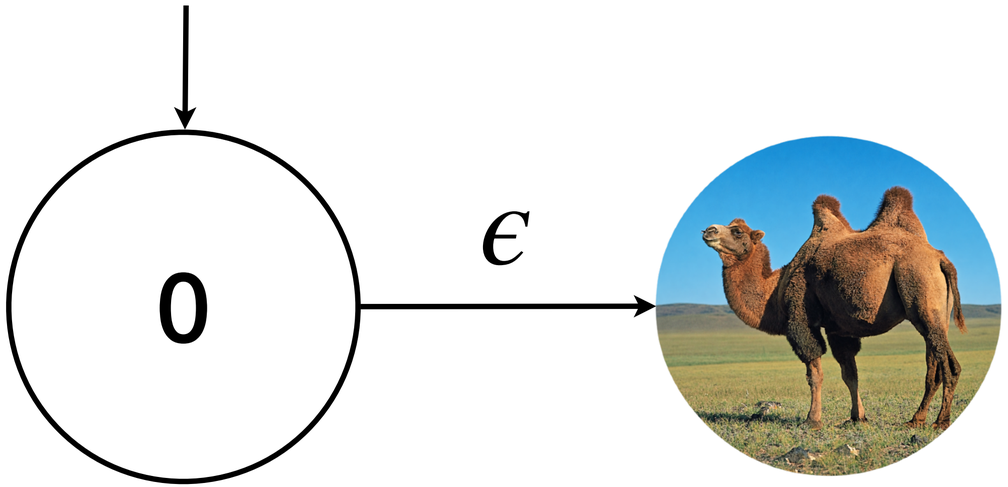
0 to OCaml in 0 steps

- **Start immediately**
  - Zero installation; runs offline in the browser
- **Learn by doing**
  - Types on hover, live errors and autocompletion
  - Immediate quiz feedback



0 to OCaml in 0 steps

- **Start immediately**
  - Zero installation; runs offline in the browser
- **Learn by doing**
  - Types on hover, live errors and autocompletion
  - Immediate quiz feedback
- **Keep the book maintainable**
  - One source for textbook and executable slides
  - Readable Git diffs and merges
  - Automatically type-check and test the code



Demo

- Running the code right in the page
  - changing it, IDE features, presenting as a slide

Demo

- Running the code right in the page
  - changing it, IDE features, presenting as a slide
- OxCaml, a performance-oriented extension of OCaml

Demo

- Running the code right in the page
  - changing it, IDE features, presenting as a slide
- OxCaml, a performance-oriented extension of OCaml
- Full Linux VM, compiled to Wasm
  - Why? Insecurity of C — buffer-overflow, use-after-free, double-free

Demo

- Running the code right in the page
  - changing it, IDE features, presenting as a slide
- OxCaml, a performance-oriented extension of OCaml
- Full Linux VM, compiled to Wasm
  - Why? Insecurity of C — buffer-overflow, use-after-free, double-free
- The joy of DOM interactivity

Tech — OCaml Platform

- **js_of_ocaml (jsoo)** — A compiler from OCaml and OxCaml to JavaScript

Tech — OCaml Platform

- **js_of_ocaml (jsoo)** — A compiler from OCaml and OxCaml to JavaScript
- OCaml compiler itself is written in OCaml
  - Platform tools too: **Merlin** —OCaml Intelligence, **OCamlFormat** — code formatting
  - *Can compile the OCaml compiler to JavaScript using jsoo!*

Tech — OCaml Platform

- **js_of_ocaml (jsoo)** — A compiler from OCaml and OxCaml to JavaScript
- OCaml compiler itself is written in OCaml
  - Platform tools too: **Merlin** —OCaml Intelligence, **OCamlFormat** — code formatting
  - *Can compile the OCaml compiler to JavaScript using jsoo!*
- **x-ocaml** — OCaml notebook as WebComponents

```
<x-ocaml>let x = 42</x-ocaml>
```

```
15  let x = 42                                          Run ▶
    val x : int = 42
```

Tech — OCaml Platform

- **js_of_ocaml (jsoo)** — A compiler from OCaml and OxCaml to JavaScript
- OCaml compiler itself is written in OCaml
  - Platform tools too: **Merlin** —OCaml Intelligence, **OCamlFormat** — code formatting
  - *Can compile the OCaml compiler to JavaScript using jsoo!*
- **x-ocaml** — OCaml notebook as WebComponents

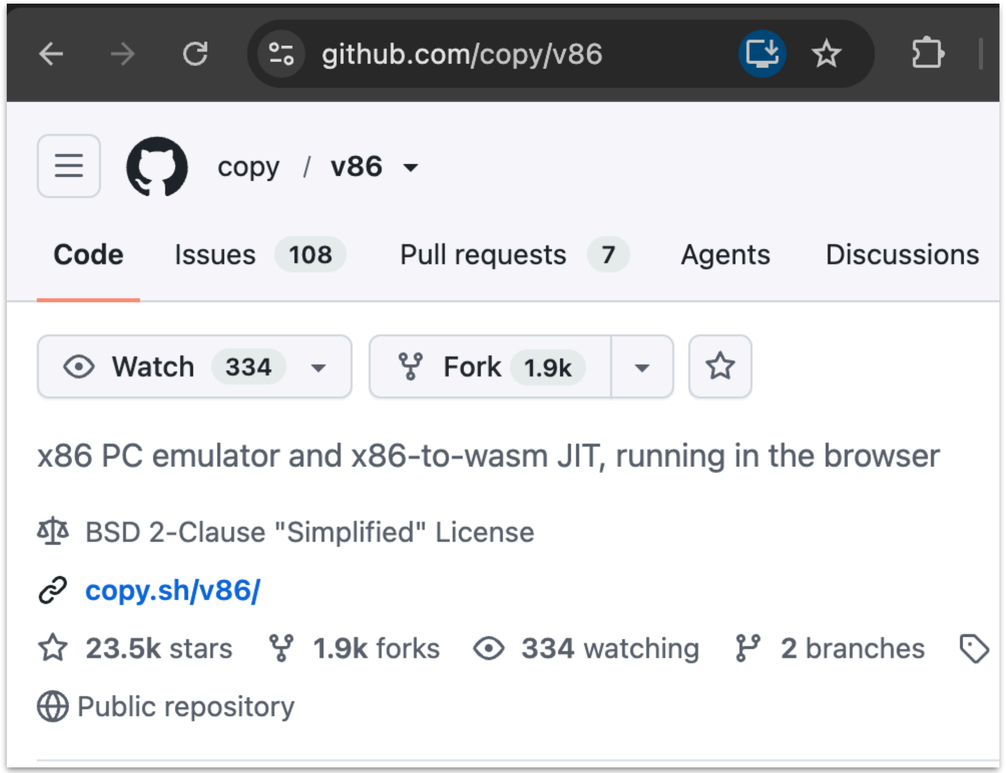
```
<x-ocaml>let x = 42</x-ocaml>
```

```
15  let x = 42                                   Run ▶
    val x : int = 42
```

Liberally licensed
Open Source software

Tech — Linux VM in the browser

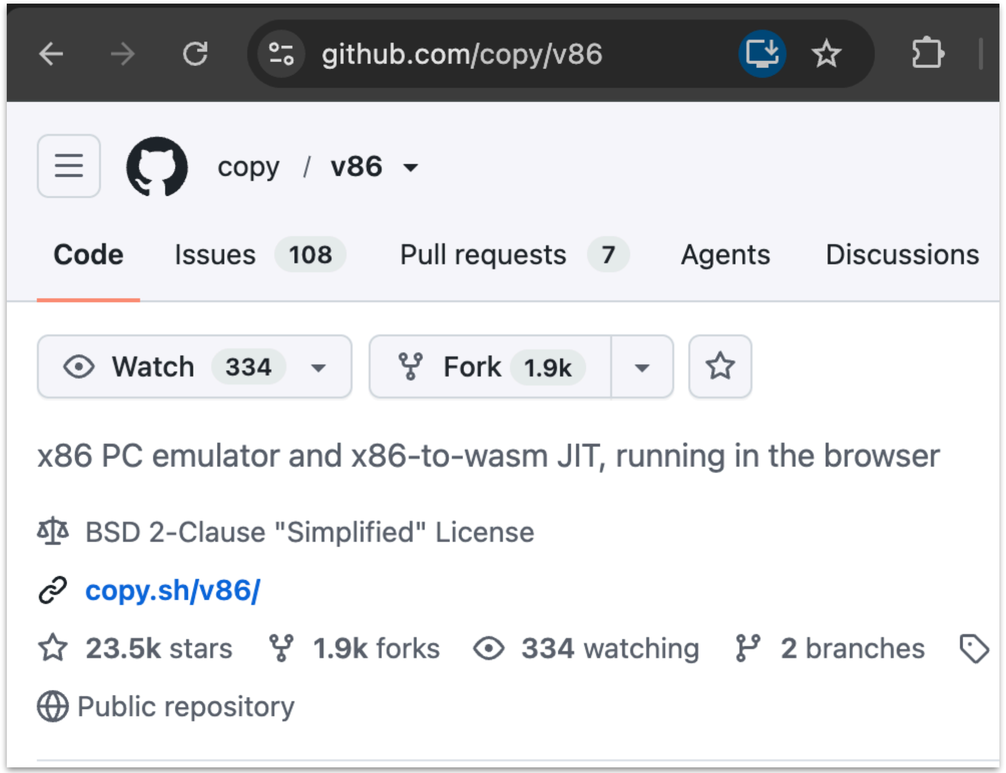
- x86 Alpine Linux image built with Docker
  - With OCaml, dune and course projects preinstalled
  - v86 JIT compiles x86 to Wasm



https://github.com/copy/v86

Tech — Linux VM in the browser

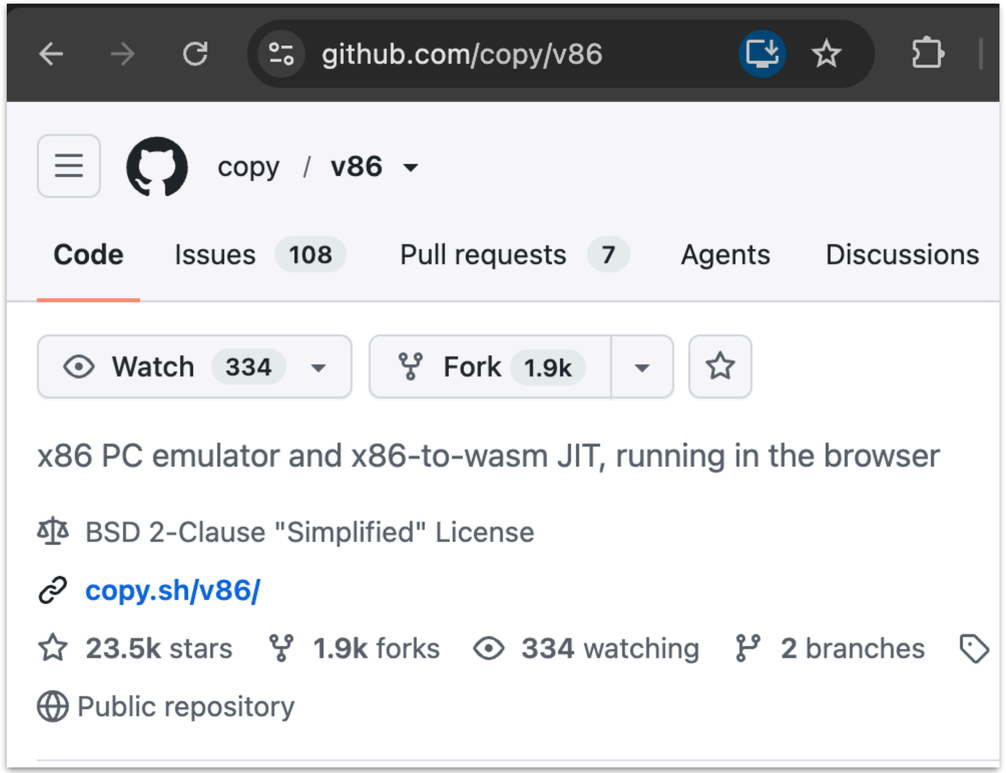
- x86 Alpine Linux image built with Docker
  - With OCaml, dune and course projects preinstalled
  - v86 JIT compiles x86 to Wasm
- **~12 MB** to a shell
  - No installation or server-side execution



https://github.com/copy/v86

Tech — Linux VM in the browser

- x86 Alpine Linux image built with Docker
  - With OCaml, dune and course projects preinstalled
  - v86 JIT compiles x86 to Wasm
- **~12 MB** to a shell
  - No installation or server-side execution
- Resumes a compressed snapshot
  - Not cold-boot
  - Loads filesystem chunks on demand



https://github.com/copy/v86

Embedded Quizzes

- Book contains embedded quizzes in lectures
  - Both MCQs and programming puzzles

Embedded Quizzes

- Book contains embedded quizzes in lectures
  - Both MCQs and programming puzzles
- Results are anonymously collected
  - Analytics available publicly

Embedded Quizzes

- Book contains embedded quizzes in lectures
  - Both MCQs and programming puzzles
- Results are anonymously collected
  - Analytics available publicly
- Feedback in both directions
  - Reader about their understanding
  - Author about their explanation
    - *Aim to revise the book based on analytics!*

Profiling Programming Language Learning

WILL CRICHTON and SHRIRAM KRISHNAMURTHI, Brown University, USA

This paper documents a year-long experiment to "profile" the process of learning a programming language: gathering data to understand what makes a language hard to learn, and using that data to improve the learning process. We added interactive quizzes to *The Rust Programming Language*, the official textbook for learning Rust. Over 13 months, 62,526 readers answered questions 1,140,202 times. First, we analyze the trajectories of readers. We find that many readers drop-out of the book early when faced with difficult language concepts like Rust's ownership types. Second, we use classical test theory and item response theory to analyze the characteristics of quiz questions. We find that better questions are more conceptual in nature, such as asking why a program does not compile vs. whether a program compiles. Third, we performed 12 interventions into the book to help readers with difficult questions. We find that on average, interventions improved quiz scores on the targeted questions by +20%. Fourth, we show that our technique can likely generalize to languages with smaller user bases by simulating our statistical inferences on small N. These results demonstrate that quizzes are a simple and useful technique for understanding language learning at all scales.

Authoring the modules

- Book is written in Markdown with Pandoc-style fenced divs

```
:::slide

## What's next

Lecture 5: **local functions and mutual recursion**.

- The `go`-inside-a-function pattern, made explicit.
- Functions that refer to each other.

:::
```

What's next

Lecture 5: **local functions and mutual recursion**.

- The `go`-inside-a-function pattern, made explicit.
- Functions that refer to each other.

Authoring the modules

- Book is written in Markdown with Pandoc-style fenced divs

````
:::quiz code id=M03-L04-q2
Write a tail-recursive `sum_of_squares : int -> int` that returns
`1*1 + 2*2 + ... + n*n` for non-negative `n`. `sum_of_squares 0 = 0`.

```ocaml
let sum_of_squares n =
  failwith "not implemented"
```

```ocaml skip
let check b m = if not b then failwith m
let () =
  check (sum_of_squares 0 = 0)    "zero";
  check (sum_of_squares 1 = 1)    "one";
  check (sum_of_squares 3 = 14)   "small: 1 + 4 + 9";
  check (sum_of_squares 5 = 55)   "five: 1 + 4 + 9 + 16 + 25";
  print_endline "all tests passed"
```
:::
````

Write a tail-recursive `sum_of_squares : int -> int` that returns `1*1 + 2*2 + ... + n*n` for non-negative `n`. `sum_of_squares 0 = 0`.

```
55  let sum_of_squares n = failwith "not implemented"
```

↺ Run ▶

Check ▾ Hide tests

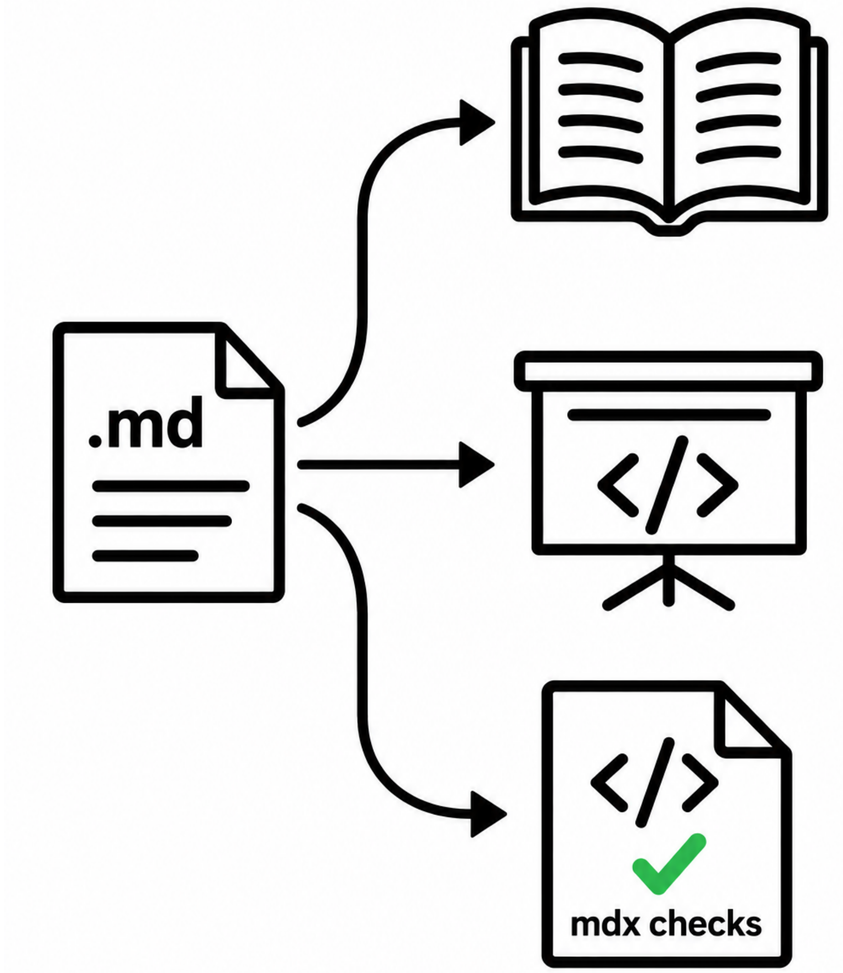
```
56  let check b m = if not b then failwith m
57
58  let () =
59    check (sum_of_squares 0 = 0) "zero";
60    check (sum_of_squares 1 = 1) "one";
61    check (sum_of_squares 3 = 14) "small: 1 + 4 + 9";
62    check (sum_of_squares 5 = 55) "five: 1 + 4 + 9 +
    16 + 25";
63    print_endline "all tests passed"
```

Run ▶

▶ Show reference solution

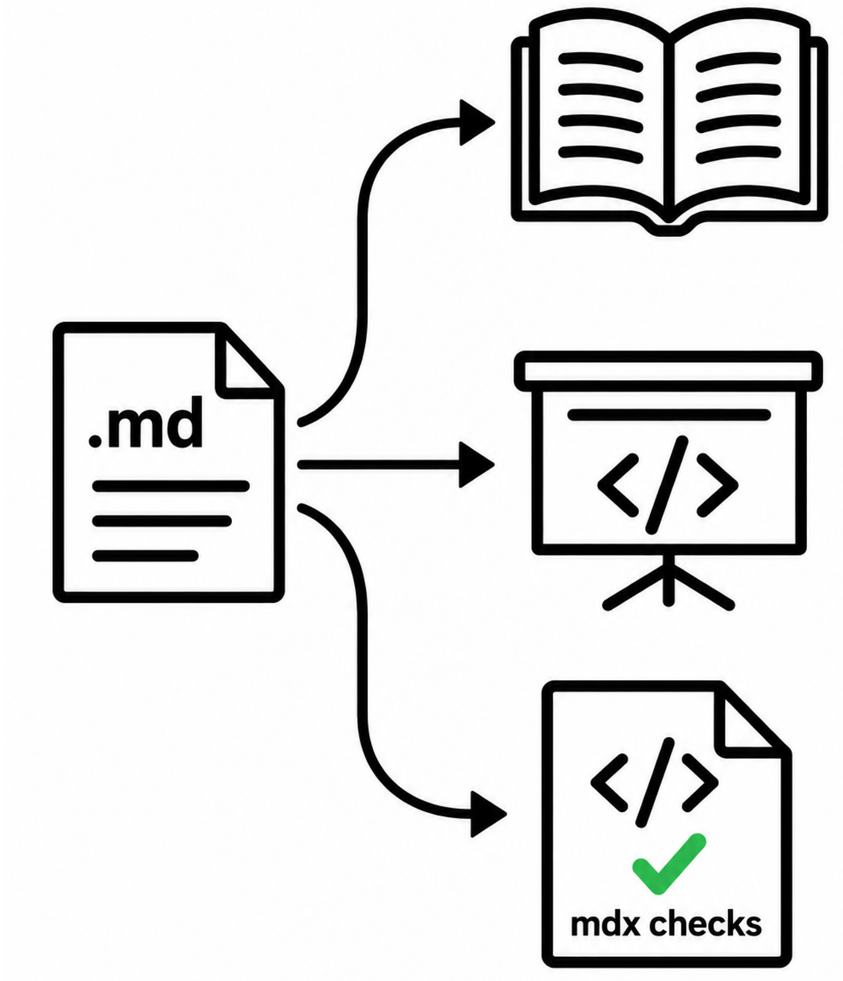
Authoring the modules

- **One markdown source for textbook and slides**
  - Helps keeps explanations and examples consistent
  - Readable Git diffs and merges



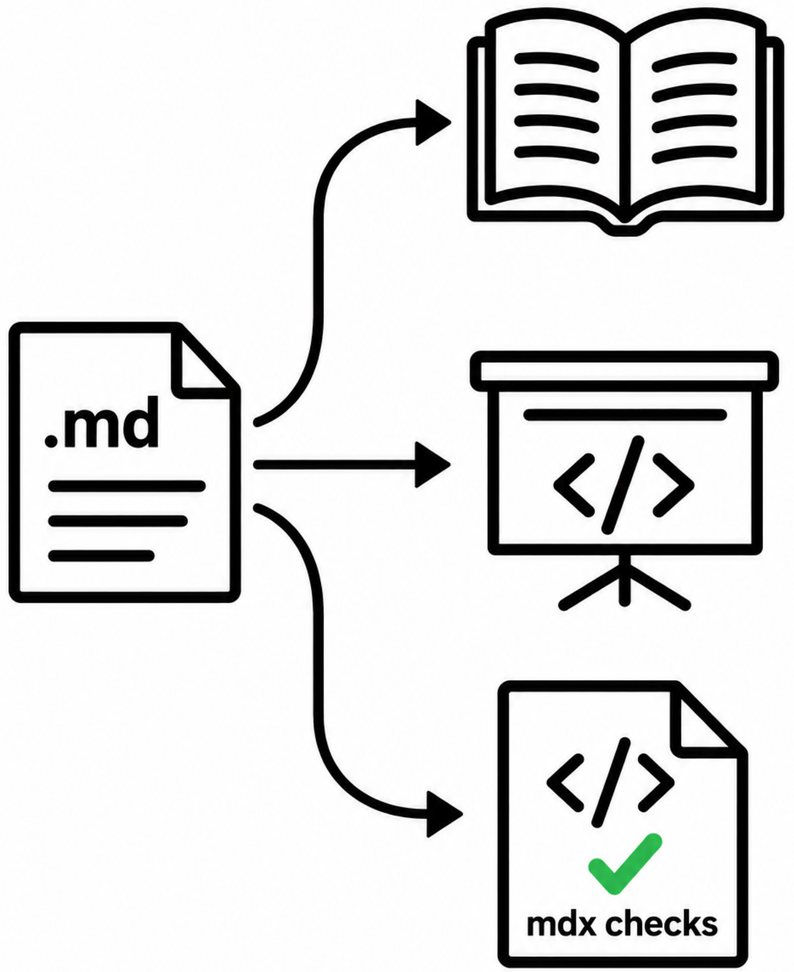
Authoring the modules

- **One markdown source for textbook and slides**
  - Helps keeps explanations and examples consistent
  - Readable Git diffs and merges
- **Supported tags**
  - slides — slide, subslide, fragment, layout — cols, col
  - quizzes — code, mcq, solution
  - vm-terminal



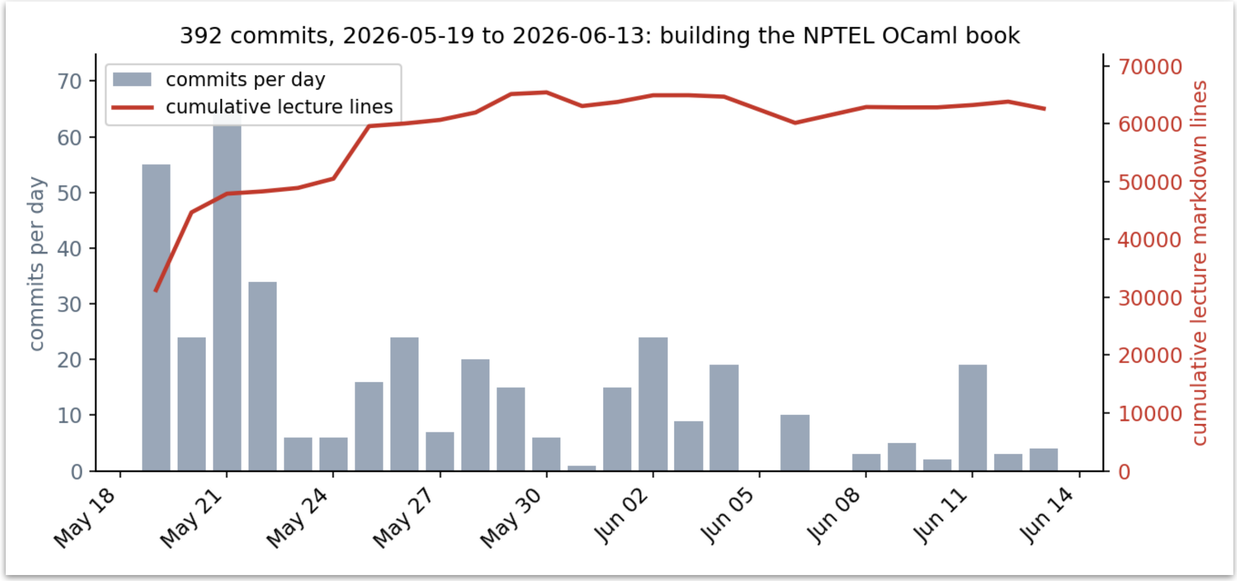
Authoring the modules

- **One markdown source for textbook and slides**
  - Helps keeps explanations and examples consistent
  - Readable Git diffs and merges
- **Supported tags**
  - slides — slide, subslide, fragment, layout — cols, col
  - quizzes — code, mcq, solution
  - vm-terminal
- ***Automated checks support review***
  - **ocaml-mdx** compiles and runs the code examples
  - Additional checks for slide overflow and broken cross-refs



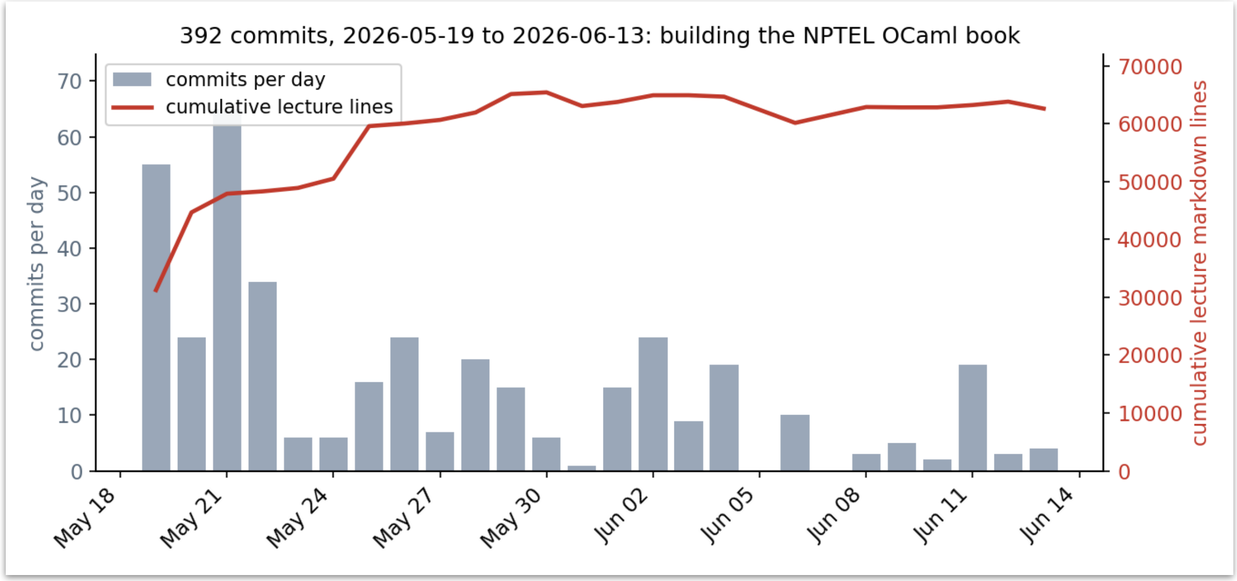
Agentic Programming

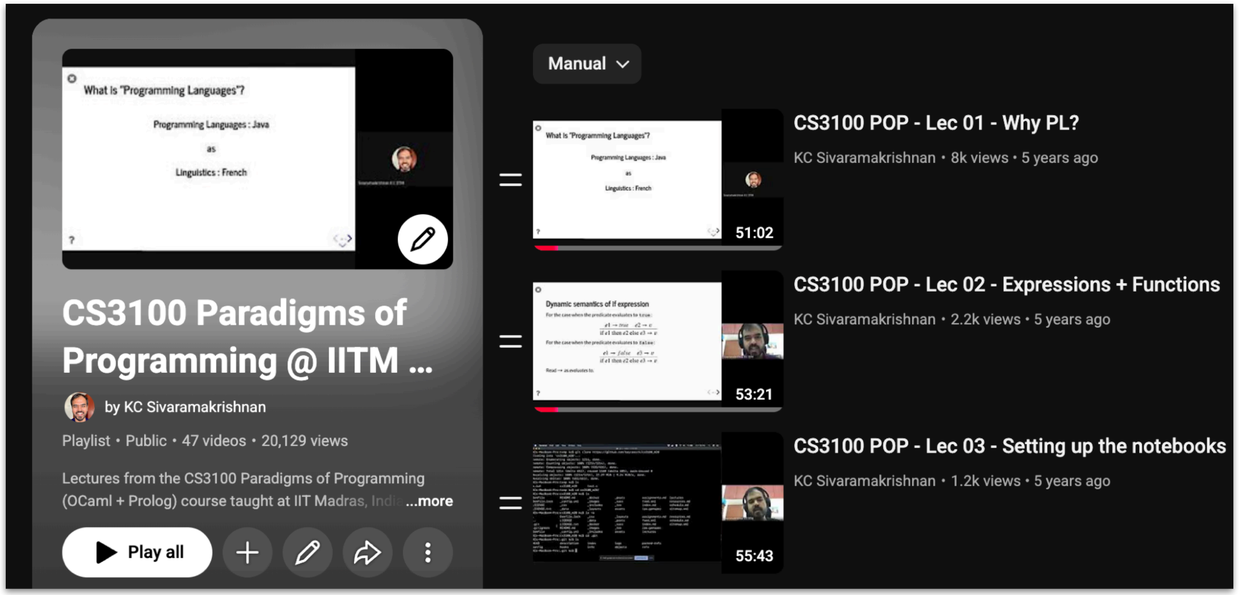
- Written in a month
  - Claude Opus 4.7+; 207M non-cached tokens
- Platform side almost entirely vibe coded.
  - ~9k LoC in OCaml, JavaScript, Shell, …



Agentic Programming

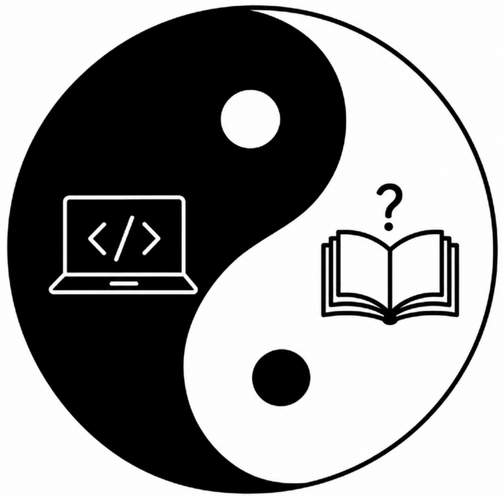
- Written in a month
  - Claude Opus 4.7+; 207M non-cached tokens
- Platform side almost entirely vibe coded.
  - ~9k LoC in OCaml, JavaScript, Shell, …
- Content is derived from IITM CS3100
  - ***60k lines of markdown***
- Agent “learnt” the material
  - ffmpeg to scene (slide) detect
  - Local Whisper model for transcribing audio
  - A small script aligns slide & transcript





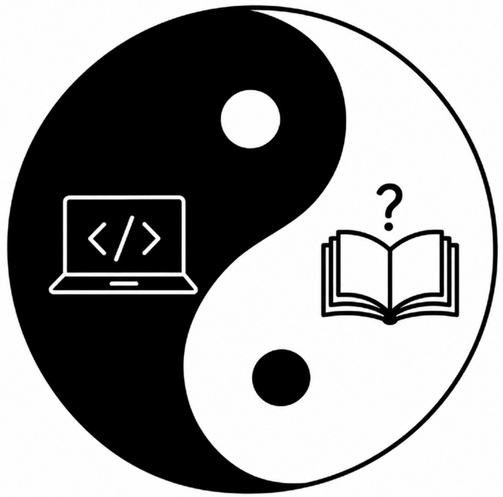
Reflections on Agentic Development

- Platform development was ***mostly good***
  - Frontend work, quizzes and analytics, etc..
  - OxCaml JS bundle size reduction from 285MB to 4MB



Reflections on Agentic Development

- Platform development was ***mostly good***
  - Frontend work, quizzes and analytics, etc..
  - OxCaml JS bundle size reduction from 285MB to 4MB
- Content writing was ***mostly bad***
  - Assumed too much about what the learner already knew
    - Used concepts before introduction; presented ideas without motivation
  - Great oracle but bad teacher
    - **Curse of Knowledge** — an expert assumes others have their specialised knowledge
    - **Pedagogical Content Knowledge** — knowing how to teach is separate from knowing



A Skill for writing chapters

- **No forward concepts**
  - Use only concepts that are currently in reader's toolbox
- **No jumps**
  - Motivate each idea, introduce using small examples
- **Slides carry the content**
  - Viewers don't read chapters first $\Longrightarrow$ slides alone must carry the narrative
- **Fresh activities**
  - Programming quizzes shouldn't repeat existing content
- **Reviews in fresh agent**
  - Prompt to assume basics and ask the agent to read preceding content

Summary

Summary

- AI is fast changing the nature of education
  - But also enables broader access to specialised knowledge
  - Gaps need educators to innovate

Summary

- AI is fast changing the nature of education
  - But also enables broader access to specialised knowledge
  - Gaps need educators to innovate
- The course seems to be going well
  - The course content is distributed under CC-BY-NC-SA

Functional Programming with OCaml
online book

Summary

- AI is fast changing the nature of education
  - But also enables broader access to specialised knowledge
  - Gaps need educators to innovate
- The course seems to be going well
  - The course content is distributed under CC-BY-NC-SA

Functional Programming with OCaml
online book

FP Launchpad

Post-bacc fellowship — Cohort 2

Deadline: Dec 10th 2026