The End of History

30 April 2021

- The Past: What do you take away from HOPL?

- The Present: What do think about your lecture? What do you think about your evaluations?

- The Future: Where do you go from here? As a PhD student? As an undergrad (incl MS)?

What is your take-away about PL after 28 lectures?

PL, the area

1. The PL research area is
   - extremely deep
   - extremely broad
   - has defined comp. sci. for 100+ years.

depth: logic arguably invented several of the essential elements of programming and the study of PLs

PL
- functions, conditionals
- types

meta
- provability vs truth
- consistency
- soundness
- reduction of fun. calls

breadth PL has also grown via programming practice and application areas.

- h.o. contracts vs interm. comp. form
- dep. types vs progr. media
± categorical sem. vs cache coherence
± static analysis vs garbage coll.
- teaching w/ Logo vs System F
- macros vs run-time (op.) sys.
± usage analysis vs proof assist.
⋮

centrality

prod. suites SW. eng. simulations

op. res. progr. comm. sys networks

comp. theory algor. systems

prog. & prog. lang.

classical AI pedagogy philosophy epistemology

modern AI (ML)

big data prob. modeling

PL is infrastructure

2. PL is often far removed from practice

$e \to e'$, $\to^+$, $\to^*$

$\vdash e = e'$

eval

$\cong$

$\Gamma \vdash e : \tau$

$\Gamma \vdash \tau \leq \tau'$

$\Theta \vdash \tau : \Gamma$

Soundness Theorem (types)
Completeness Theorem (contracts)

How do researchers know that their work eventually, somehow applies to what the working developer does?

3. PL is often far ahead of practice

- functional programming (McCarthy '61)
  ⇝ more and more practical work moves into this direction

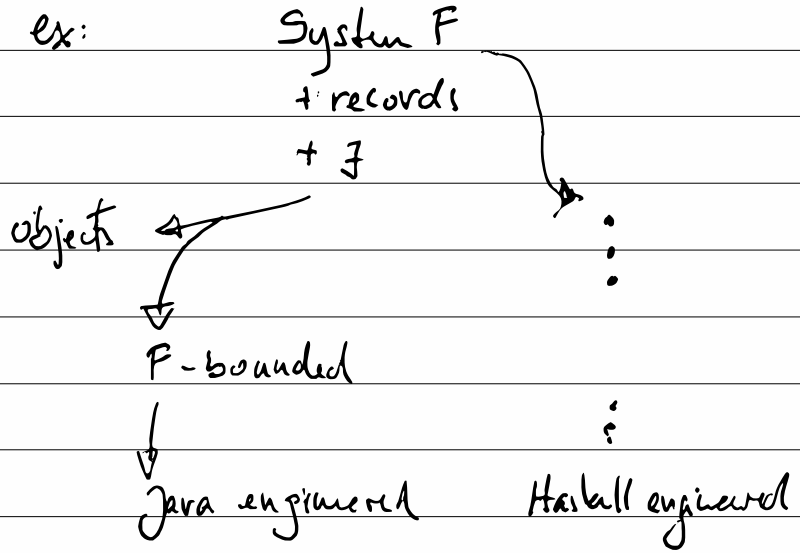
- macros (Lisp '65, Cardelli 83–87)
  ⇝ Rust, Scala, Template Haskell are definitely inspired by these ideas; or borrow them

- garbage collection (Lisp '65)
  ⇝ Java (98) finally made it clear that GC is better than programmers managing pointers

- System F types & polym. abstr. '73
  ⇝ Java (98) & Haskell (90s) put this form of types in or ten map

4. Research is incremental

start w/ a "breakthrough idea"
explore in different directions
settle on the ideas



Popper, The Logic of Scientific Discovery
Kuhn, The Structure of Scientific Revolutions
Stevens, The Knowledge Machine

5. A lot of research becomes invisible

many researchers explore small steps, off the branches and twigs of research trees

- "research bureaucrats"
- truly failed attempts
- invisible fabric

ex: the evolution from System F to sophisticated type systems for OOPL

1990s uncountable number of papers on how to encode OO systems into typed λ systems

and

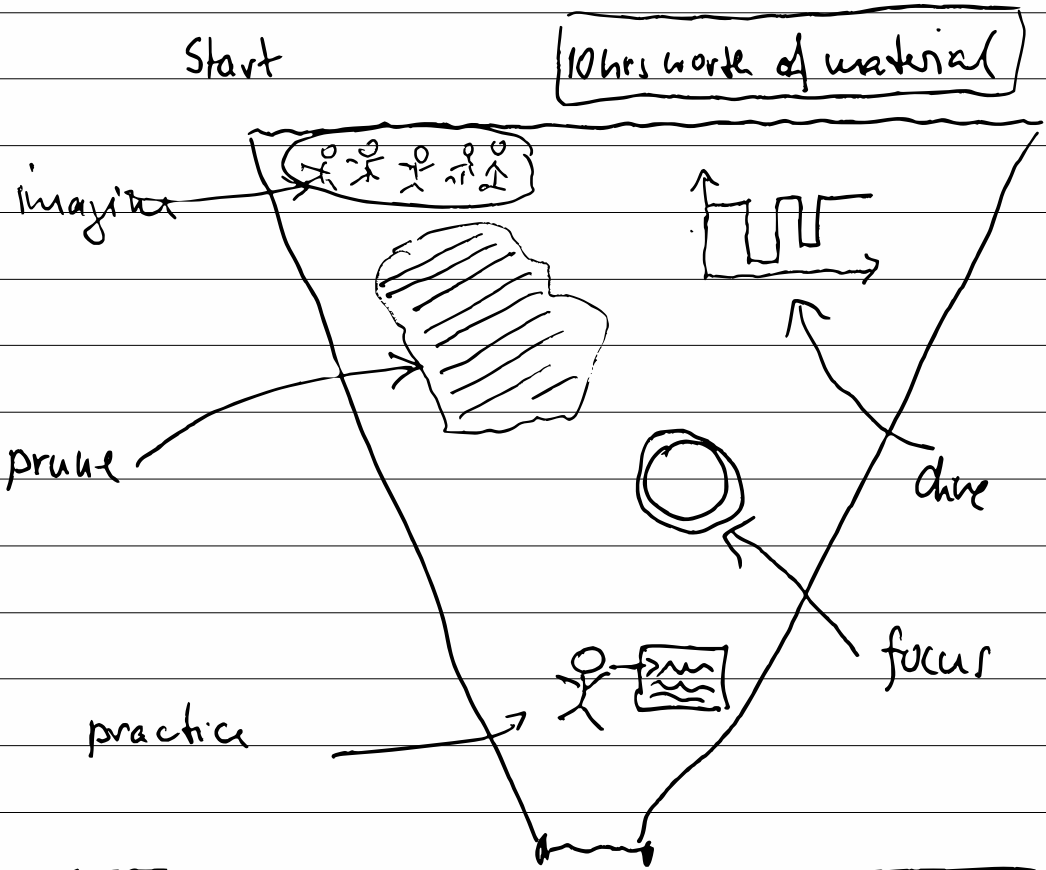
OO calculi like $\lambda_{\&}$ and types for them

What do you now think about

- YOUR LECTURE

- YOUR EVALUATIONS

1. Lecturing



Your audience
choose & focus
pick levels of detail
practice your words

2. Evaluating

You will evaluate others.

- your evaluations as students are usually worthless (except for s...d XYZ deans)

- like everything, you must practice

- learn from teaching & comparing how others evaluate the exact same scene

GRADES

What does your HOPL experience tell you about your future

- as an undergrad
- as a PhD student

You thought PL was cool
heard there's more to learn
read some books & papers

You thought you'd go for a PhD
in this world

You are overwhelmed

1. A course does not tell
you whether you should
go into a PhD program
or whether you're able to
finish.

2. Take a deep breath.
Read more about your ♡s.
Reflect on your skills,
background, personality

"I am ready for the hottest topic you have."

- Research is fashion-driven.
  popularity
  citation counts & cliques
  "hot topics"

- Research is reality-driven.
  popularity
  download counts & comments
  "hot topics"

My opinion:
  ∀ hot topics
    ∃ many people
      offer some contribution
      & solution
  and only 1 or 2 become lucky
  winners

"I am a math genius."
"I can build systems standing on my head."

go to math.
go to industry.

MY OPINION:

A good PL research program proves theorems.

A good piece of PL research is inspired by practice but doesn't seem to be practical now.

The best don't worry but do good things.

This is it.