

CSE 230

Programming

Languages



Tony Hoare

Turing Award Lecture 1980

*“There are two ways of constructing software.
One way is to make it so simple,
that there are obviously no deficiencies,
The other way is to make it so complicated
that there are no obvious deficiencies.”*

Goal: Obviously No Deficiencies

Readable

Goal: Obviously No Deficiencies

Reusable

Goal: Obviously No Deficiencies

Modifiable

Goal: Obviously No Deficiencies

Predictable

Goal: Obviously No Deficiencies

Checkable

Goal: Obviously No Deficiencies

Yes, but how?

Goal: Obviously No Deficiencies

Functional Programming(?)

Functional Programming ?

No Assignment.

No Mutation.

No Loops.

So, Who Uses FP ?

PL Researchers.

Functional Programming ?

Readable
Reusable
Modifiable
Predictable
Checkable

Parallelizable



So, Who Uses FP ?



MapReduce

So, Who Uses FP ?

Google

 TensorFlow

So, Who Uses FP ?



Microsoft[®]

F#

So, Who Uses FP ?



twitter

Scala

So, Who Uses FP ?



Erlang

So, Who Uses FP ?



Verse

So, Who Uses FP ?

CSE 230

CSE 230: Medium of Instruction



Why Haskell ?

Bleeding edge PL.

Why Haskell ?

Beautiful.

Why Haskell ?

Blows Your Mind.

Why Haskell ?



Alan Perlis
Epigrams In Programming

*“A language that doesn't affect how you think
about programming, isn't worth knowing”*

Why Haskell ?

Fun.

CSE 230 : Outline

Readable

Reusable

1. FP & Abstraction

Modifiable

Predictable

Checkable

2. Types & Analysis

CSE 230 : Outline

- The Lambda Calculus
- Types and Haskell
- Advanced FP
- Formal Verification

CSE 230 : Resources

Lectures, assignments, piazza, OH:

<https://ucsd-cse230.github.io/fa23>

Please sign yourselves up at PIAZZA & Github

CSE 230 : Grading

[40%] : 5 Programming Assignments

[30%] : 2 x Midterms

[30%] : Final Project

?