CSE 230
Programming Languages
Please “mute” your Mic
By default, unless I call on you

If you have a question
“raise hand” or type question in chat

Lectures split into ~ 3-4 segments
~15 mins “lecture” + ~5 mins “exercise”
“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

Readable

Readable
Goal: Obviously No Deficiencies

Reusable

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
Parallelizable
Predictable
Checkable
So, Who Uses FP?

Google

MapReduce
So, Who Uses FP?

Google

TensorFlow
So, Who Uses FP?

Microsoft

F#
So, Who Uses FP?

Scala
So, Who Uses FP?

Facebook

Erlang
So, Who Uses FP?

WhatsApp

Erlang
So, Who Uses FP?

CSE 230
CSE 230: Medium of Instruction

Haskell
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

1. FP & Abstraction
   Readable
   Reusable

2. Types & Analysis
   Modifiable
   Predictable
   Checkable
All materials (lectures, assignments, piazza) at:

https://ucsd-cse230.github.io/fa21

Please sign yourselves up at PIAZZA
CSE 230 : Lectures on Zoom

Recordings available on CANVAS

Each lecture is about 3-4 segments

- ~15 mins “RJ: lecture/Q&A”
- ~5 mins “You: exercise”
CSE 230 : Grading

[70%] : 4~5 Programming Assignments

[30%] : Final Project

[5%] : Extra Credit for top-15 Piazzars