CSE 230
Programming Languages
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
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
Why Haskell?

Bleeding edge PL.
Why Haskell?

Beautiful.
Why Haskell?

Blows Your Mind.
“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
   - Modifiable
   - Predictable
   - Checkable

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

https://ucsd-cse230.github.io/sp20

Please sign yourselves up at PIAZZA
Recordings available on CANVAS

Each lecture is about 3 segments

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

[75%] : 4~5 Programming Assignments

[25%] : Final Exam

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