

# Principles of Programming Languages

## Lecture 09 Subtyping Bounded Universal and Existential Types

Albert-Ludwigs-Universität Freiburg

Peter Thiemann

University of Freiburg, Germany

[thiemann@informatik.uni-freiburg.de](mailto:thiemann@informatik.uni-freiburg.de)

25 June 2018



UNI  
FREIBURG



## 1 Subtyping Quantified Types

# Recall Subtyping for Function Types



$$\frac{\vdash T_1 <: S_1 \quad \vdash S_2 <: T_2}{\vdash S_1 \rightarrow S_2 <: T_1 \rightarrow T_2}$$

- **contravariant** in argument position: flip types
- **covariant** in result position: keep order of types

# Bounded Universal Types



## Syntax

$$T ::= \dots \mid \forall X <: T. T$$



# Bounded Universal Types

## Syntax

$$T ::= \dots \mid \forall X <: T. T$$

New: May have subtyping assumptions for type variables  $X$

$$\frac{X <: T \in \Gamma}{\Gamma \vdash X <: T}$$

# Bounded Universal Types

## Syntax

$$T ::= \dots \mid \forall X <: T. T$$

New: May have subtyping assumptions for type variables  $X$

$$\frac{X <: T \in \Gamma}{\Gamma \vdash X <: T}$$

Consider  $\forall$  as a function

$$\frac{\Gamma \vdash T_1 <: S_1 \quad \Gamma, X <: T_1 \vdash S_2 <: T_2}{\Gamma \vdash \forall X <: S_1. S_2 <: \forall X <: T_1. T_2}$$

## Syntax

$$T ::= \dots \mid \exists X <: T. T$$

## Syntax

$$T ::= \dots \mid \exists X <: T. T$$

## Covariant in bound of exists

$$\frac{\Gamma \vdash S_1 <: T_1 \quad \Gamma, X <: S_1 \vdash S_2 <: T_2}{\Gamma \vdash \exists X <: S_1. S_2 <: \exists X <: T_1. T_2}$$





- Benjamin Pierce. Types in Programming Languages. MIT Press 2002.