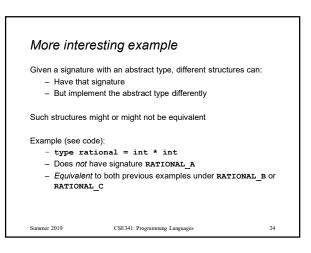


- Equivalent under RATIONAL_B or RATIONAL_C
 - Different invariants, but same properties
 - Essential that type rational is abstract

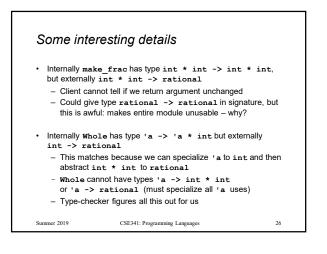
Summer 2019

CSE341: Programming Languages

23



Summa S



Can't mix-and-match module bindings Modules with the same signatures still define different types So things like this do not type-check! . ational1.toString(Rational2.make_frac(9,6)) . ational3.toString(Rational2.make_frac(9,6)) . Different modules have different internal invariants . Different modules have different internal invariants . bifferent modules have different type definitions . ational1.rational looks like Rational2.rational, . ational3.rational is int*int not a datatype!