## Source Code NiLang IR Julia Language begin begin $n \leftarrow zero(T)$ Julia @routine begin n = zero(T)n += abs(x)@assignback ((PlusEq)(abs))(n, x) true $n \leftarrow zero(T)$ Compile Translate Preprocess end end n += abs(x)v!.re += log(n)@assignback ((PlusEq)(log))(y!.re, n) true end y!.im += angle(x)@assignback ((PlusEq)(angle))(y!.im, x) true y!.re += log(n)begin begin v!.im += angle(x)@assignback ((MinusEq)(abs))(n, x) true n -= abs(x)~@routine (NiLangCore.deanc)(n, zero(T)) $n \rightarrow zero(T)$ end end Reverse LLVM begin begin $n \leftarrow zero(T)$ Julia n = zero(T)n += abs(x)(assignback ((PlusEq)(abs))(n, x) trueCompile Translate end end @assignback ((MinusEq)(angle))(y!.im, x) true y!.im -= angle(x)@assignback ((MinusEq)(log))(y!.re, n) true v!.re -= log(n)begin begin @assignback ((MinusEq)(abs))(n, x) true n -= abs(x)(NiLangCore.deanc)(n, zero(T)) $n \rightarrow zero(T)$ end end