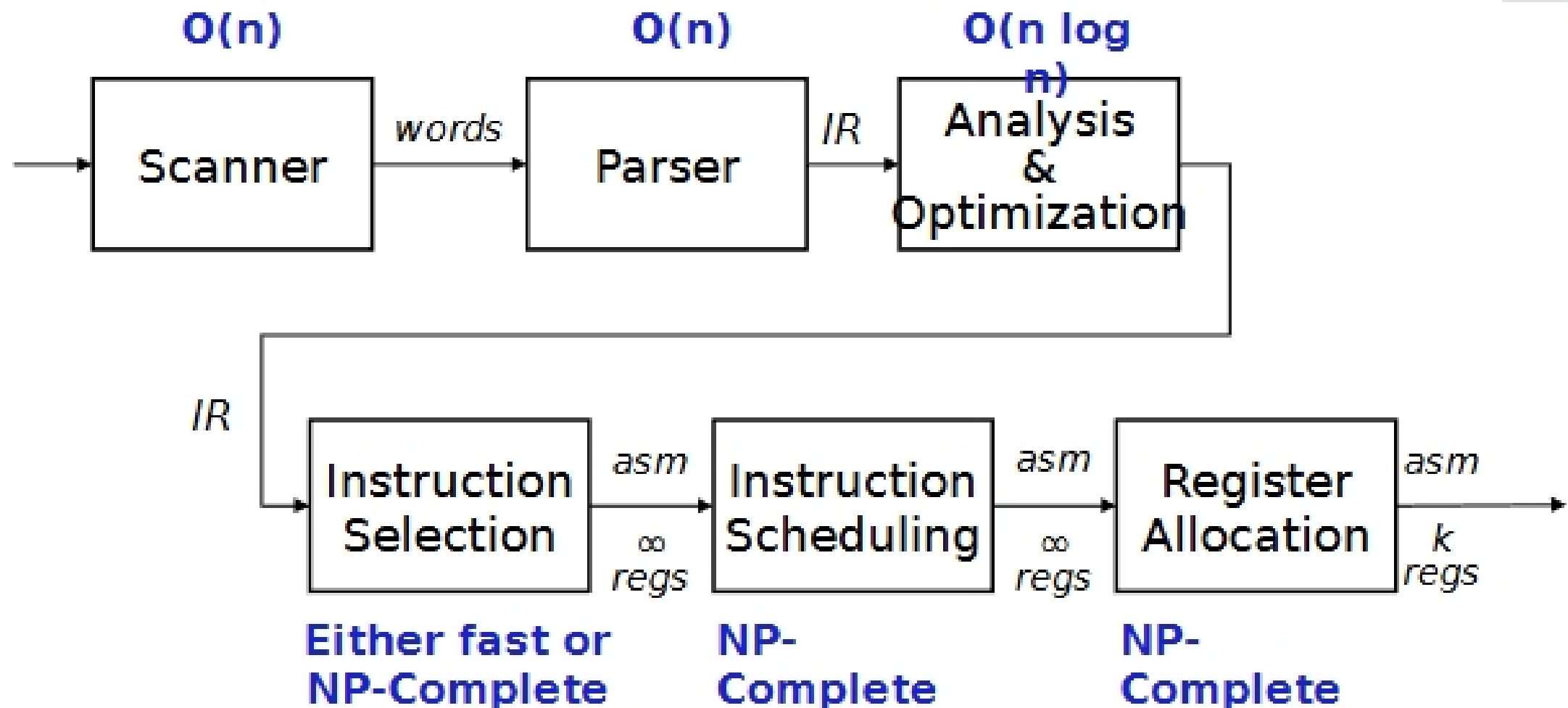




Introduction to Code Generation

Copyright 2003, Keith D. Cooper, Ken Kennedy & Linda Torczon, all rights reserved.

Structure of a Compiler

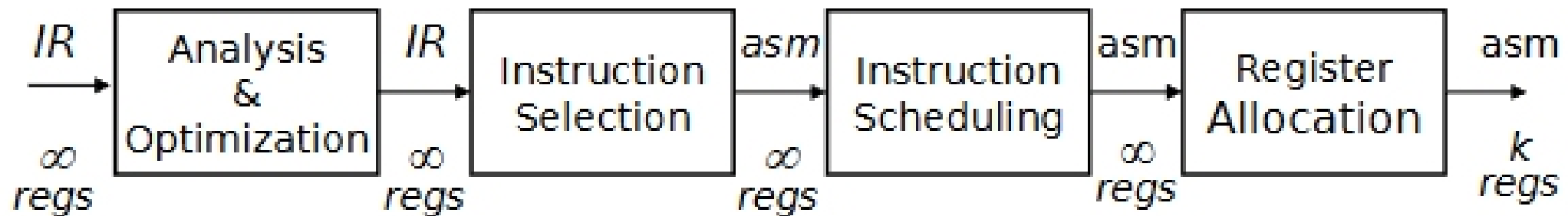


A compiler is a lot of fast stuff followed by some hard problems

- The hard stuff is mostly in **optimization** and **code generation**
- For superscalars, its allocation & scheduling that count

Structure of a Compiler

For the rest of CISC672, we assume the following model



- Selection is fairly simple (problem of the 1980s)
- Allocation & scheduling are complex
- Operation placement is not yet critical (*unified register set*)

What about the IR ?

- Low-level, RISC-like IR called **ILO**
- Has “enough” registers
- **ILOC** was designed for this stuff

Branches, compares, & labels
Memory tags
Hierarchy of loads & stores
Provision for multiple ops/cycle