

## 四、中间代码生成

### (3. 控制流语句的翻译 I)

魏恒峰

hfwei@nju.edu.cn

2023 年 05 月 10 日



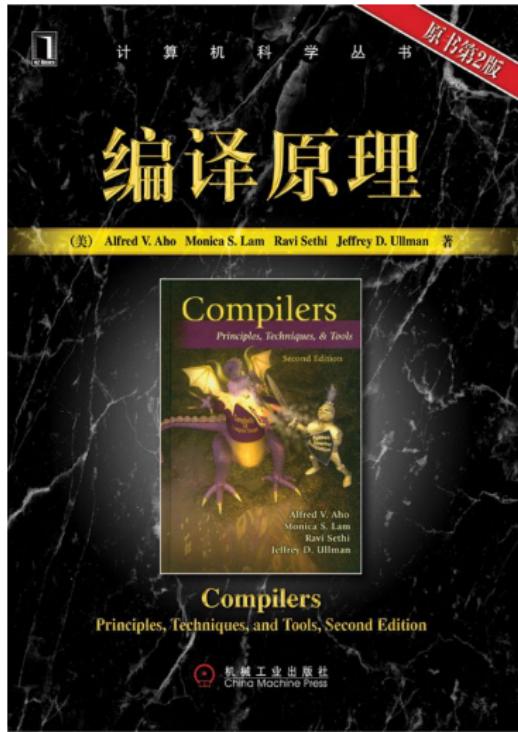
## Control.g4

产生式
$P \rightarrow S$
$S \rightarrow \text{assign}$
$S \rightarrow \text{if} ( B ) S_1$
$S \rightarrow \text{if} ( B ) S_1 \text{ else } S_2$
$S \rightarrow \text{while} ( B ) S_1$
$S \rightarrow S_1 S_2$

产生式
$B \rightarrow B_1 \uparrow\downarrow B_2$
$B \rightarrow B_1 \&\& B_2$
$B \rightarrow ! B_1$
$B \rightarrow E_1 \text{ rel } E_2$
$B \rightarrow \text{true}$
$B \rightarrow \text{false}$



本讲内容**非常重要**, 但**颇有难度**, 需要多多思考



虽然有**多种**讲法，教材偏偏采用了让初学者**望而生畏**的那一种。

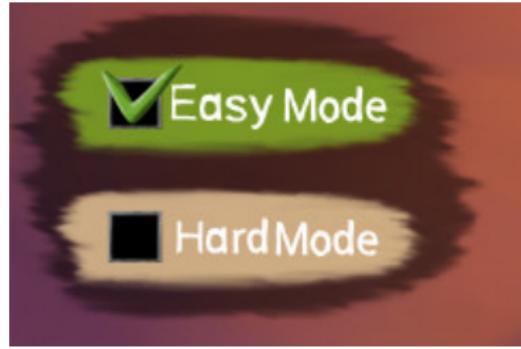
“We will overcome all difficulties.” (我们有困难要上)



“If we don't have enough difficulties, we'll create them!”  
(没困难, 我们创造困难还要上)

关键问题：为什么要“创造困难”？

生成更短、更高效的代码





心中有“树”(语法分析树)

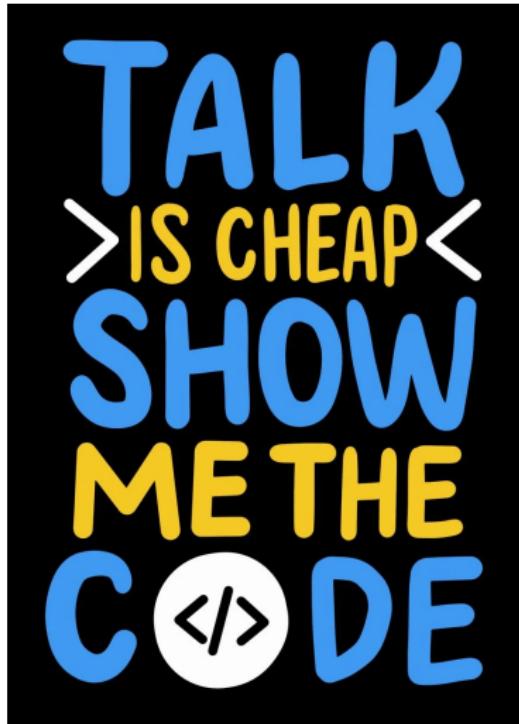
# 分工 合作



为布尔表达式  $B$  计算逻辑值 (假设保存在临时变量  $t1$  中)

**if**、**while** 等语句根据  $B$  的结果改变控制流

**if** ( $B$ )  $S_1$



## CodeGenVisitor.java

## 选择实现方式: Listeners, **Visitors**, Attributed Grammar

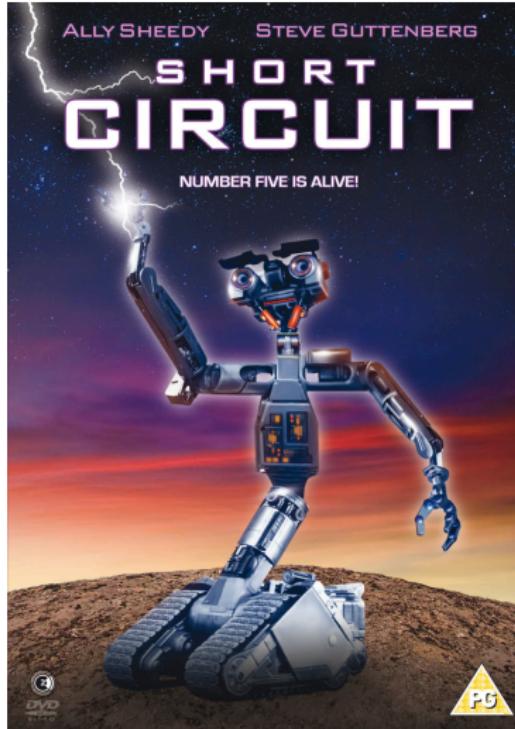
产生式
$P \rightarrow S$
$S \rightarrow \text{assign}$
$S \rightarrow \text{if} ( B ) S_1$
$S \rightarrow \text{if} ( B ) S_1 \text{ else } S_2$
$S \rightarrow \text{while} ( B ) S_1$
$S \rightarrow S_1 S_2$

产生式
$B \rightarrow B_1 \sqcup \sqcap B_2$
$B \rightarrow B_1 \&\& B_2$
$B \rightarrow ! B_1$
$B \rightarrow E_1 \text{ rel } E_2$
$B \rightarrow \text{true}$
$B \rightarrow \text{false}$

及时输出生成的中间代码, 避免频繁的字符串拼接操作

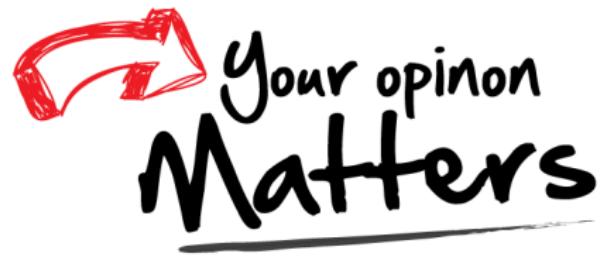


How to implement “break” statements?



How to implement “Short-Circuit” evaluation?

# Thank You!



Office 926

hfwei@nju.edu.cn