

Arith : A Simple Language

grammar  $\{ e ::= n \mid e + e \mid e * e \mid e - e \}$

$n \in \mathbb{Z}$

$n$  → values

$e + e \mid e * e \mid e - e$  → Expressions but not values

$e_1$   $e_2$

$(2 + 3) * (8 - 4)$

$e_1$   $e_2$

## Dynamic Semantics of Arith

Rules to Explain how to evaluate expressions in Arith

Premise <sub>1</sub>	Premise <sub>2</sub>	...	Premise <sub>n</sub>
<hr/>			
Conclusion			

"Semantic rule"

$e_1 \rightarrow v_1$	$e_2 \rightarrow v_2$	$v = v_1 + v_2$
<hr/>		
$e_1 + e_2 \rightarrow v$		

→ Evaluation to

C-like

2+ variable

Program  $P ::= S$

Statement  $S ::= x := e \mid S ; S$

$M$   
     $x = 1 ;$   
 $M$      $y = 2 ;$   
 $M$      $x = x + y ;$   
     $y = x - y ;$   
     $z = x - y ;$

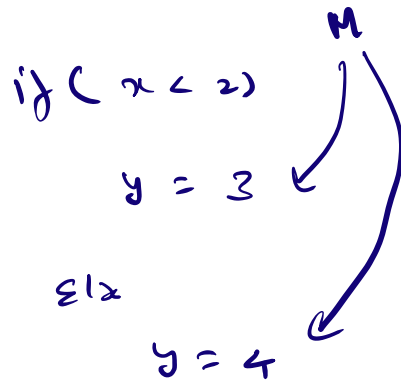
Swapping program in C-like

$M_{final}$   
Dynamic Semantics C-like

$M : Var \rightarrow Value$

$M \vdash e \rightarrow v$   
-----  
 $(M, (x := e)) \rightarrow (v, M[x \rightarrow v])$

$(x + y) \rightarrow ?$   
-----  
 $x = (x + y)$

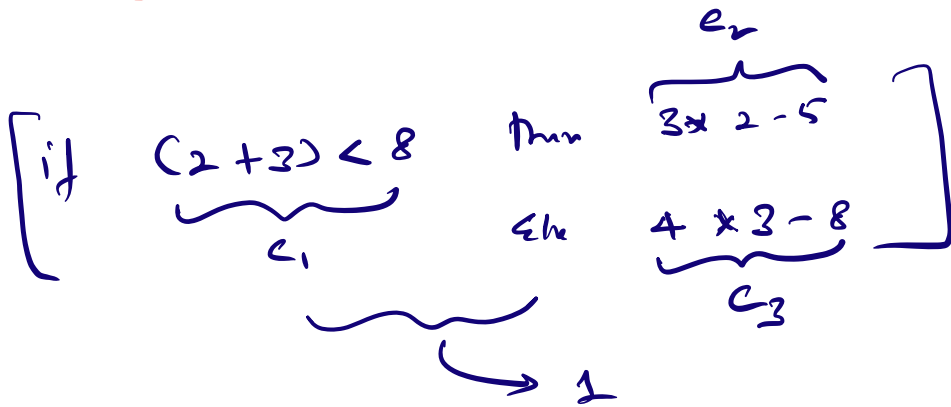


## Arith Boal

$n \in \mathbb{Z}$        $b \in B$

$e ::= n \mid e + e \mid e * e \mid e - e \mid e / e \mid$   
 $\quad\quad\quad cve \mid rel$

$\text{if } e \text{ Then } e \text{ Else } e$


$$\begin{array}{l} e_1 \rightarrow \text{true} \quad c_2 \rightarrow v \\ \hline \text{if } e_1 \text{ then } c_2 \text{ else } \varepsilon_3 \rightarrow v \end{array}$$

$$c_1 \rightarrow \text{false} \quad c_3 \rightarrow \text{true}$$

---

$$\text{if } c_1 \text{ then } c_2 \text{ else } c_3 \rightarrow \text{true}$$