

CS 421 --- Big Step Semantics Activity

Manager	Keeps team on track	
Recorder	Records decisions	
Reporter	Reports to class	
Reflector	Assesses team performance	

The Rules

Expression Rules

$$\frac{}{\langle i, \sigma \rangle \Downarrow_e i} \text{CONST, when } i \text{ is a integer.}$$

$$\frac{}{\langle u, \sigma \rangle \Downarrow_e v} \text{VAR, if } u := v \in \sigma.$$

$$\frac{\langle e_1, \sigma \rangle \Downarrow_e v_1 \quad \langle e_2, \sigma \rangle \Downarrow_e v_2}{\langle e_1 \oplus e_2, \sigma \rangle \Downarrow_e v_1 \oplus v_2} \text{ARITH}$$

Boolean Rules

$$\frac{}{\langle i, \sigma \rangle \Downarrow_b b} \text{CONST, when } b \text{ is true or false .}$$

$$\frac{}{\langle u, \sigma \rangle \Downarrow_b v} \text{VAR, if } u := v \in \sigma.$$

$$\frac{\langle e_1, \sigma \rangle \Downarrow_e v_1 \quad \langle e_2, \sigma \rangle \Downarrow_e v_2}{\langle e_1 \sim e_2, \sigma \rangle \Downarrow_b v_1 \sim v_2} \text{REL}$$

Statement Rules

$$\frac{}{\langle \text{skip}, \sigma \rangle \Downarrow \sigma} \text{SKIP}$$

$$\frac{\langle e, \sigma \rangle \Downarrow_e v}{\langle x := e, \sigma \rangle \Downarrow \sigma[x := v]} \text{ASSIGN}$$

$$\frac{\langle S_1, \sigma \rangle \Downarrow \sigma' \quad \langle S_2, \sigma' \rangle \Downarrow \sigma''}{\langle S_1; S_2, \sigma \rangle \Downarrow \sigma''} \text{SEQ}$$

$$\frac{\langle B, \sigma \rangle \Downarrow_b \text{true} \quad \langle S_1, \sigma \rangle \Downarrow \sigma'}{\langle \text{if } B \text{ then } S_1 \text{ else } S_2 \text{ fi}, \sigma \rangle \Downarrow \sigma'} \text{IF}_1$$

$$\frac{\langle B, \sigma \rangle \Downarrow_b \text{false} \quad \langle S_2, \sigma \rangle \Downarrow \sigma'}{\langle \text{if } B \text{ then } S_1 \text{ else } S_2 \text{ fi}, \sigma \rangle \Downarrow \sigma'} \text{IF}_2$$

$$\frac{\langle B, \sigma \rangle \Downarrow_b \text{false}}{\langle \text{while } B \text{ do } S \text{ od}, \sigma \rangle \Downarrow \sigma} \text{WHILE}_1$$

$$\frac{\langle B, \sigma \rangle \Downarrow_b \text{true} \quad \langle S, \sigma \rangle \Downarrow \sigma' \quad \langle \text{while } B \text{ do } S \text{ od}, \sigma' \rangle \Downarrow \sigma''}{\langle \text{while } B \text{ do } S \text{ od}, \sigma \rangle \Downarrow \sigma''} \text{WHILE}_2$$

Understanding the Rules

Problem 1) Consider the judgment $\langle x + y, \{x := 10, y := 20\} \rangle \Downarrow_e 30$. Explain the meaning of the different syntactic components:

- $\langle x + y, \{x := 10, y := 20\} \rangle$
- \Downarrow_e

Problem 2) There are some rules that come in almost identical pairs, for example, there are two versions of the VAR rule. What is the difference between them, and why do we need both?

Using the Rules

One use of the rules is to build proof trees. Consider the following proof tree for the initial configuration $\langle x + y + z, \{x := 10, y := 20, z := 30\} \rangle$.

$$\frac{\frac{\overline{\langle x, \sigma \rangle \Downarrow_e 10} \text{ VAR} \quad \frac{\overline{\langle y, \sigma \rangle \Downarrow_e 20} \text{ VAR}}{\langle x \oplus y, \sigma \rangle \Downarrow_e 30} \text{ ARITH} \quad \frac{\overline{\langle z, \sigma \rangle \Downarrow_e 30} \text{ VAR}}{\langle 10 + 20 + 30, \sigma \rangle \Downarrow_e 60} \text{ ARITH}}{\langle 10 + 20 + 30, \sigma \rangle \Downarrow_e 60} \text{ ARITH}$$

Problem 3) The ARITH rule has a $e_1 \oplus e_2$ and $v_1 \oplus v_2$ in it, but the proof tree only has something corresponding to $e_1 \oplus e_2$. What happened to $v_1 \oplus v_2$?

Problem 4) Suppose we told you that addition were right associative instead of left associative. What would the proof tree look like in that case?

Problem 5) Write the proof tree for the following program.

`t:=a; a:=b; b:=t, {a:=5, b:=10}>`

Problem 6) Write the proof tree for one of the following programs:

`<if x>y then m:=x-y else m:=y-x fi, {x:=10, y:=30}>`

or

`<while x > 1 do x:=x/2 od, {x:=2}>`

Make your own rules!

Problem 7)

Write a rule to explain the when B S statement. It executes S only if B is true.

Problem 8)

Write a rule for do S while B od. It is like while, but executes S at least one time.

Big Step Semantics Activity--- Reflector's Report

Manager	Keeps team on track	
Recorder	Records decisions	
Reporter	Reports to Class	
Reflector	Assesses team performance	

1. What was a strength of your team's performance for this activity?

2. What could you do next time to increase your team's performance?

3. What insights did you have about the activity or your team's interaction today?