



CS 311O

The Substitution Model

Prof. Clarkson

Fall 2019

Today's music: Substitute by The Who

CLICKER QUESTION 1

Review

Previously in 3110:

- interpreter for tiny calculator language
- lexing into tokens
- parsing into abstract syntax tree (AST)
- evaluation based on single steps

Today:

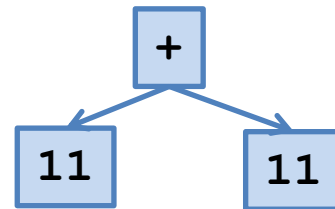
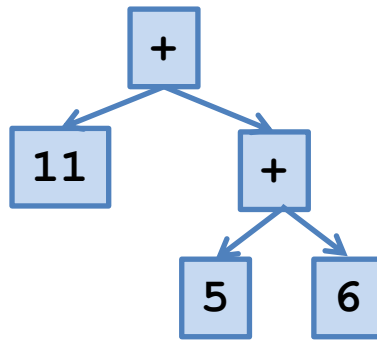
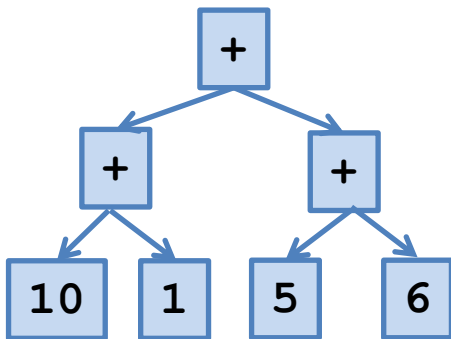
- finish evaluation
- extend to a bigger language: let expressions
- substitution model: a way to think about evaluation of `let`

Review: calculator language BNF

$$\begin{aligned} e &::= i \\ &\quad | e1 \text{ bop } e2 \\ &\quad | (e) \end{aligned}$$
$$\text{bop} ::= + \mid *$$
$$i ::= \textit{integers}$$

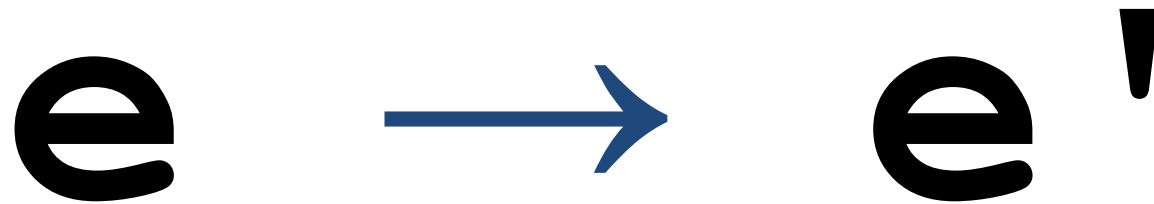
Review: evaluation strategy

- An expression e takes a single *step* to a new expression e' by simplifying some subexpression
- Expression keeps stepping until it reaches a *value*
- Values never step further



CLICKER QUESTION 2

FORMAL DYNAMIC SEMANTICS

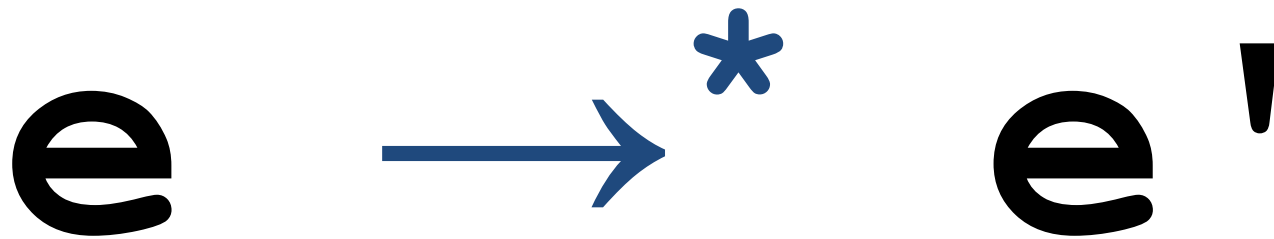


single-step relation

the `step` function we implemented



values never step



multi-step relation

the `eval` function we implemented

CLICKER QUESTION 3

Inductively defined relation

$$e1 + e2 \rightarrow e1' + e2$$

if $e1 \rightarrow e1'$

$$v1 + e2 \rightarrow v1 + e2'$$

if $e2 \rightarrow e2'$

$$v1 + v2 \rightarrow i$$

if i is the result of primitive operation $v1 + v2$

LET EXPRESSIONS

Let expressions

$e ::= i$
| $e1 \text{ bop } e2$
| (e)
| x
| $\text{let } x = e1 \text{ in } e2$

$\text{bop} ::= + \mid *$

$i ::= \textit{integers}$

$x ::= \textit{identifiers}$

Let semantics

$\text{let } x = e1 \text{ in } e2 \rightarrow \text{let } x = e1' \text{ in } e2$
if $e1 \rightarrow e1'$

$\text{let } x = v1 \text{ in } e2 \rightarrow$
 $e2 \text{ with } v1 \text{ substituted for } x$

$$e\{v/x\}$$

means e with v substituted for x

Let semantics

$\text{let } x = e1 \text{ in } e2 \rightarrow \text{let } x = e1' \text{ in } e2$
if $e1 \rightarrow e1'$

$\text{let } x = v1 \text{ in } e2 \rightarrow e2\{v1/x\}$

Defining substitution: the easy parts

Nothing to do for integers:

$$i \{v/x\} = i$$

Just keep going through operations:

$$(e1 + e2) \{v/x\} = (e1 \{v/x\}) + (e2 \{v/x\})$$

Variables are where substitution really happens:

$$x \{v/x\} = v$$

$$y \{v/x\} = y$$

Defining substitution: let

$$(\text{let } y = e1 \text{ in } e2) \{v/x\}$$
$$=$$
$$\text{let } y = (e1 \{v/x\}) \text{ in } (e2 \{v/x\})$$

Do substitute in
binding.

e.g.,
let $x = 1$ in
(let $y = x$ in y)

$$(\text{let } x = e1 \text{ in } e2) \{v/x\}$$
$$=$$
$$\text{let } x = (e1 \{v/x\}) \text{ in } e2$$

Stop substituting
in body.

e.g.,
let $x = 1$ in
(let $x = 2$ in x)

gets even trickier in the presence of functions: see textbook re. capture-avoiding substitution

If expressions

$e ::= x \mid i \mid b$
 $\mid e1 \text{ bop } e2$
 $\mid \text{let } x = e1 \text{ in } e2$
 $\mid \text{if } e1 \text{ then } e2 \text{ else } e3$

$\text{bop} ::= + \mid * \mid <=$

If semantics

`if e1 then e2 else e3 → if e1' then e2 else e3`
`if e1 → e1'`

`if true then e2 else e3 → e2`

`if false then e2 else e3 → e3`

Upcoming events

- [last night] A5 due

This is not a substitute.

THIS IS 3110