6.1120 Fall 2025: Recitation 4

Phase 2 Overview

Kosi Nwabueze

September 22, 2025

Massachusetts Institute of Technology

Announcements

1. **Phase 1**
   - Reminder that Phase 1 is due **10pm** today (Monday, September 22).
   - Please submit to Gradescope *as soon as possible* to make sure there are no build issues.
2. **Phase 2**
   - Phase 2 has officially released (as of 12pm).
   - Phase 2 will be due at **10pm** on **Friday, October 3**.
   - *Please start early!*
3. **Miniquiz**
   - There is a miniquiz due tomorrow at 11:59pm.
   - Forgot to release it on Friday, so I extended the time.

Interpreter

```
// cat arith.mit
sum = 1 + 1;
print("1 + 1 = " + sum);
```

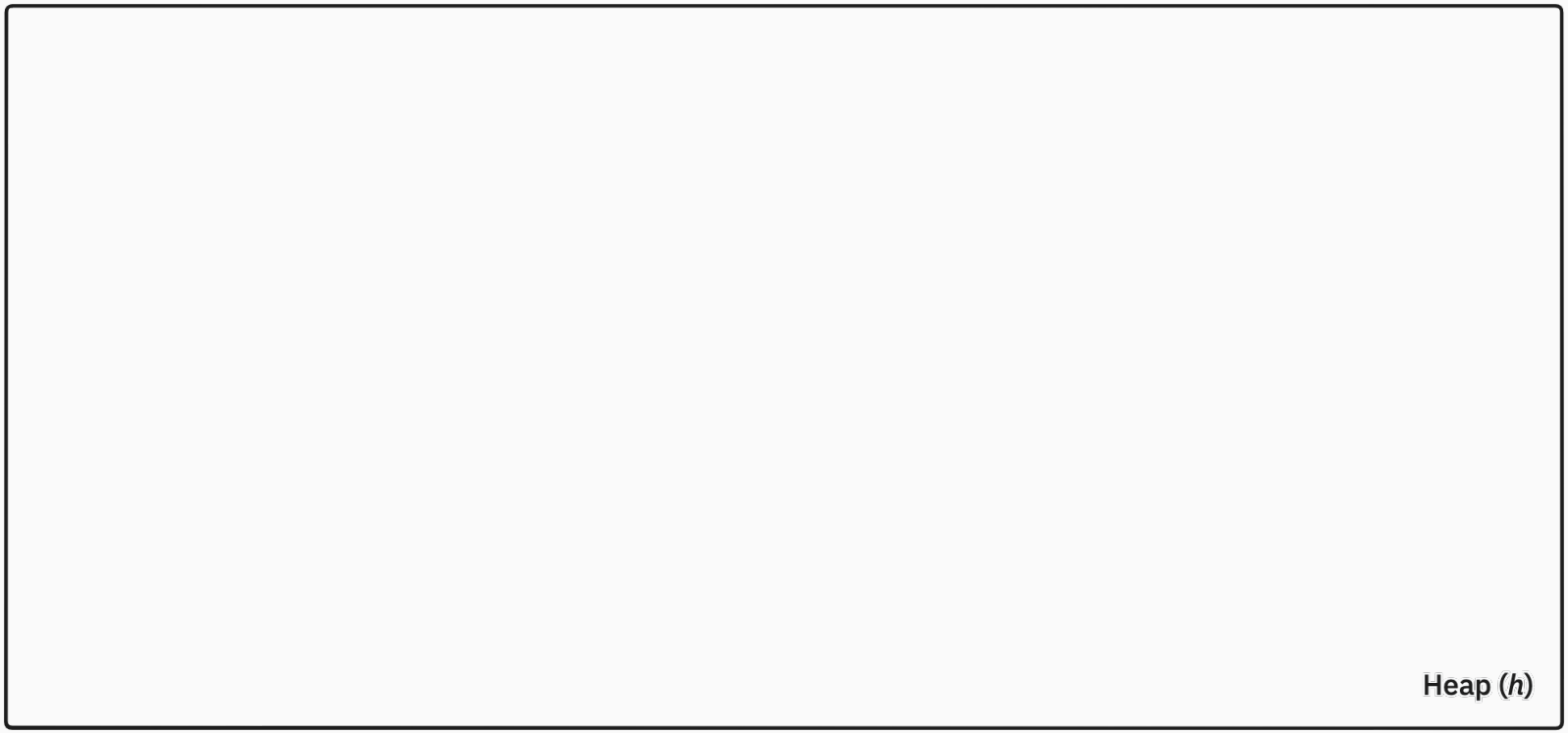
```
// ./run.sh interpret arith.mit
1 + 1 = 2
```

Deliverables

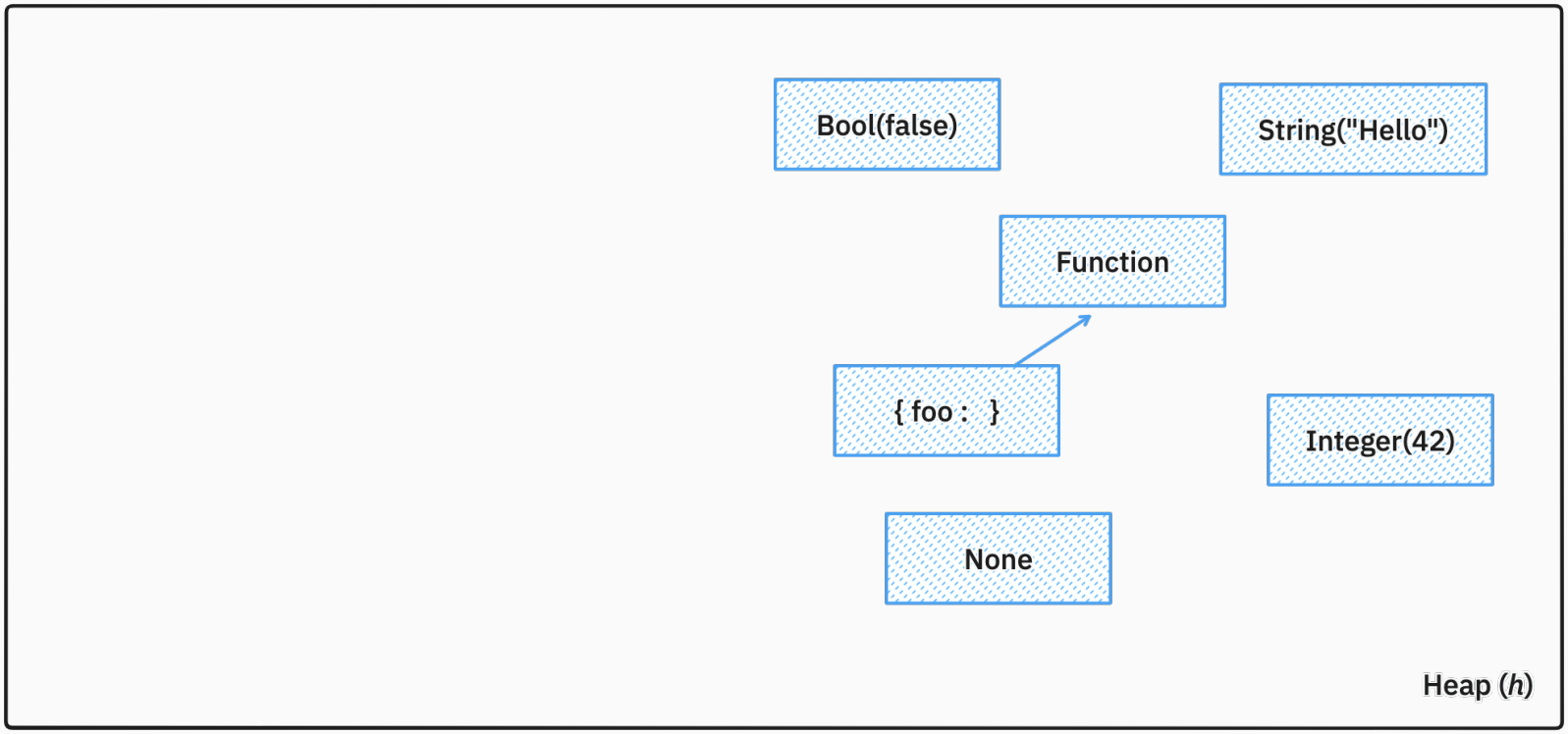
All the following deliverables are due, **October 3** at **10pm**.

1. **Implementation**
   - Must conform to the semantics described in the language specifications.
   - Must be able to build your code using `./build.sh` (again submit to Gradescope early, esp. if you changed the build system).
2. **Short report** (up to 3 paragraphs).
3. **Ten additional test cases**.

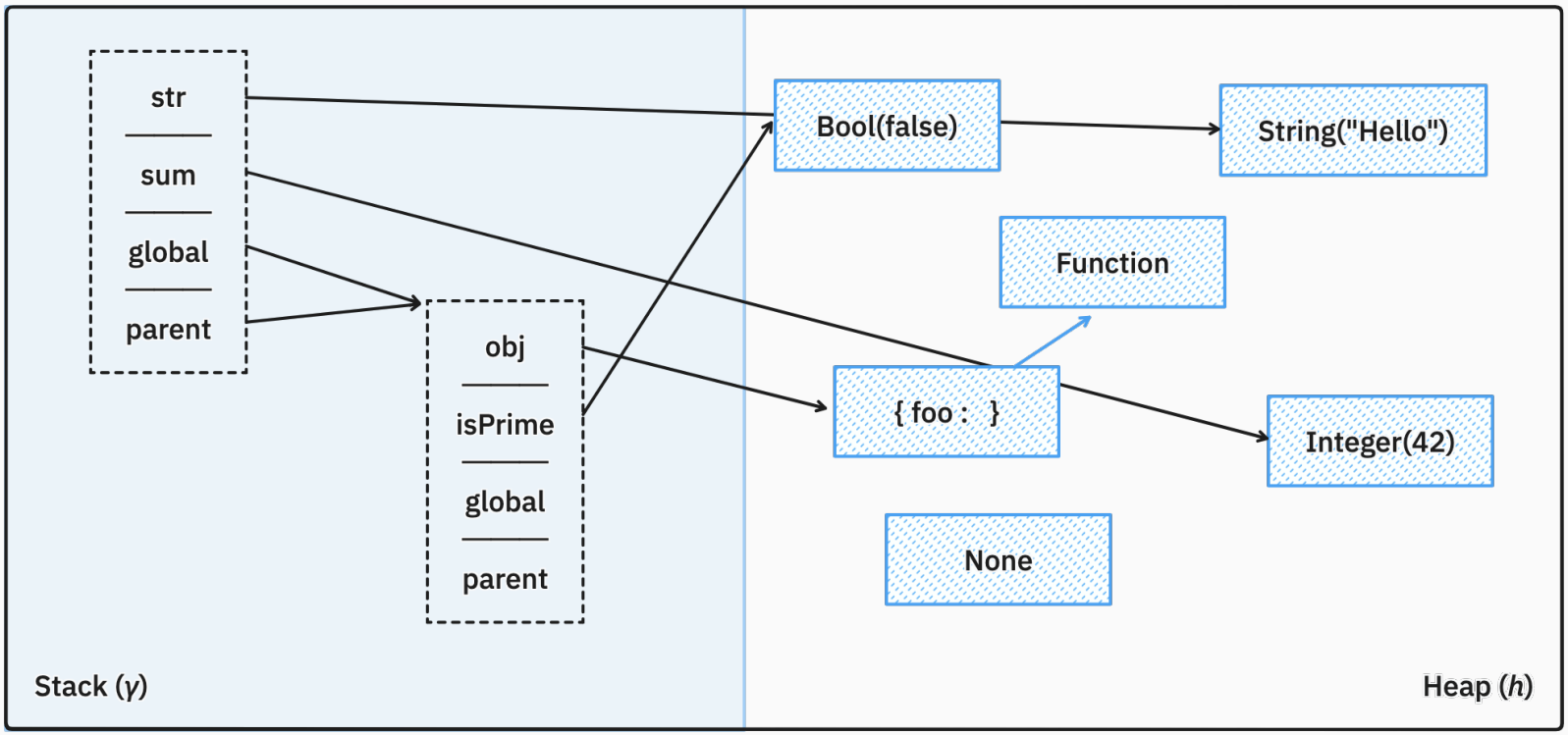
Semantics



Semantics



Semantics


str
sum
global
parent
obj
isPrime
global
parent
Bool(false)
String("Hello")
Function
{ foo : }
Integer(42)
None
Stack (γ)
Heap (h)

Semantics

The style of semantics covered in 6.1120 is sometimes called *operational semantics*. This style is very popular for constructing proofs from inference rules which talk about a program's execution.

In particular, MITScript defines its semantics using **small-step operational semantics**.

Semantics (Statements)

$$\textsc{Var}\ \frac{(\gamma; a_1, h, e) \to (h', v) \qquad \text{lookup-write}(a_1, h', x) = a_2 \qquad \sigma = h'(a_2) \qquad a_3 \notin \text{dom}(h') \qquad \sigma' = \sigma[x : a_3]}{(\gamma; a_1, h, x = e) \to h'[a_3 : v][a_2 : \sigma']}$$

$$\textsc{IfFalse}\ \frac{(\gamma, h, e) \to (h', \text{Bool(false)}) \qquad (\gamma, h', s_2) \to \eta}{(\gamma, h, \text{if } e \ \ s_1 \text{ else } s_2) \to \eta}$$

$$\textsc{IfTrue}\ \frac{(\gamma, h, e) \to (h', \text{Bool(true)}) \qquad (\gamma, h', s_1) \to \eta}{(\gamma, h, \text{if } e \ \ s_1 \text{ else } s_2) \to \eta}$$

$$\text{S}\ \frac{(\gamma, h, s_1) \to h' \qquad (\gamma, h', s_2) \to \eta}{(\gamma, h, s_1; s_2) \to \eta} \qquad \text{S-R}\ \frac{(\gamma, h, s_1) \to \text{Return}(h', v)}{(\gamma, h, s_1; s_2) \to \text{Return}(h', v)}$$

Semantics (Expressions)

$$\textsc{IntConst}\ \frac{}{(\gamma, h, n) \to (h, \text{Integer}(n))}$$

$$\textsc{ArithOp}\ \frac{(\gamma, h, e_1) \to (h', \text{Integer}(n_1)) \qquad (\gamma, h', e_3) \to (h'', \text{Integer}(n_2))}{(\gamma, h, e_1 \text{ op } e_2) \to (h'', \text{Integer}(n_1 \text{ op } n_2))}$$

$$\textsc{VarRead}\ \frac{\text{lookup-read}(a_1, h, x) = a_2 \qquad h(a_2) = v}{(\gamma; a_1, h, x) \to (h, v)}$$

$$\textsc{Function}\ \frac{\begin{array}{c} m = \{\text{ctx} : a,\ \text{formal} : x,\ \text{body} : s\} \\ a_f \notin \text{dom}(h) \qquad h' = h[a_f : m] \end{array}}{(\gamma; a, h, \text{fun } x\ s) \to (h', \text{Function}(a_f))}$$

Example

(global x; $x = 2$;)