[redacted]

Store domain models a computer store as a mapping from the ide

Figure 5.1

I. Truth Values

Domain $t \in Tr = \mathbb{B}$

Operations

$true, false : Tr$

$not : Tr \rightarrow Tr$

II. Identifiers

Domain $i \in Id =$ Identifier

III. Natural Numbers

Domain $n \in Nat = \mathbb{N}$

Operations

$zero, one, \cdots : Nat$

$plus : Nat \times Nat \rightarrow Nat$

$equals : Nat \times Nat \rightarrow Tr$

IV. Store

Domain $s \in Store = Id \rightarrow Nat$

Operations

$newstore : Store$

$newstore = \lambda i.\ zero$

$access : Id \rightarrow Store \rightarrow Nat$

$access = \lambda i.\ \lambda s.\ s(i)$

$update : Id \rightarrow Nat \rightarrow Store \rightarrow Store$

$update = \lambda i.\lambda n.\lambda s.[\,i \mapsto n\,]s$

Abstract syntax:

P ∈ Program
C ∈ Command
E ∈ Expression
B ∈ Boolean-expr
I ∈ Identifier
N ∈ Numeral

P ::= C.
C ::= C_1;C_2 | **if** B **then** C | **if** B **then** C_1 **else** C_2 | I:=E | **diverge**
E ::= E_1+E_2 | I | N
B ::= E_1=E_2 | ¬ B

Semantic algebras:
(defined in Figure 5.1)

Valuation functions:

P: Program→*Nat*→$Nat_\perp$

$\mathbf{P}[\![C.]\!] = \lambda n.\ \text{let } s = (update\ [\![A]\!]\ n\ newstore) \text{ in}$
$\qquad \text{let } s' = \mathbf{C}[\![C]\!]s \text{ in } (access\ [\![Z]\!]\ s')$

C: Command→$Store_\perp$→$Store_\perp$

$\mathbf{C}[\![C_1;C_2]\!] = \underline{\lambda} s.\ \mathbf{C}[\![C_2]\!](\mathbf{C}[\![C_1]\!]s)$
$\mathbf{C}[\![\textbf{if } B \textbf{ then } C]\!] = \underline{\lambda} s.\ \mathbf{B}[\![B]\!]s \rightarrow \mathbf{C}[\![C]\!]s\ [\!]\ s$
$\mathbf{C}[\![\textbf{if } B \textbf{ then } C_1 \textbf{ else } C_2]\!] = \underline{\lambda} s.\ \mathbf{B}[\![B]\!]s \rightarrow \mathbf{C}[\![C_1]\!]s\ [\!]\ \mathbf{C}[\![C_2]\!]s$
$\mathbf{C}[\![I{:=}E]\!] = \underline{\lambda} s.\ update\ [\![I]\!]\ (\mathbf{E}[\![E]\!]s)\ s$
$\mathbf{C}[\![\textbf{diverge}]\!] = \underline{\lambda} s.\ \perp$

E: Expression→*Store*→*Nat*

$\mathbf{E}[\![E_1+E_2]\!] = \lambda s.\ \mathbf{E}[\![E_1]\!]s\ plus\ \mathbf{E}[\![E_2]\!]s$
$\mathbf{E}[\![I]\!] = \lambda s.\ access\ [\![I]\!]\ s$
$\mathbf{E}[\![N]\!] = \lambda s.\ \mathbf{N}[\![N]\!]$

B: Boolean-expr→*Store*→*Tr*

$\mathbf{B}[\![E_1=E_2]\!] = \lambda s.\ \mathbf{E}[\![E_1]\!]s\ equals\ \mathbf{E}[\![E_2]\!]s$
$\mathbf{B}[\![\neg B]\!] = \lambda s.\ not(\mathbf{B}[\![B]\!]s)$

N: Numeral→*Nat* (*omitted*)