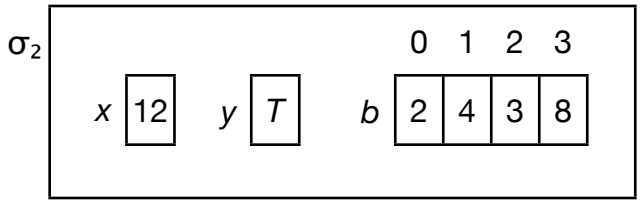
Types, Expressions, and States
CS 536 Lecture 3, Wed Jan 18, 2012

A. Why?
- Expressions represent values in programming languages, relative to a state.
- Types describe common properties of sets of values.
- States describe memory; an expression has a value relative to a state; a predicate is satisfied or unsatisfied relative to a state.

B. Outcomes
At the end of today, you should
- Be able to recognize expressions we'll be using in our language.
- Be able to recognize a state is and know what it means for a state to be proper.
- Be able to evaluate an expression relative to a state.
- Be able to check for satisfaction (truth) of a predicate in a state.

C. Questions
1. Which of the following expressions are legal or illegal according to the syntax we're using? Assume x, y, z are integer variables, b is an array, and f and g are functions on integers.
   a. $(x > y \ ? \ x : y)$
   b. $(x < y \ ? \ -1 : (x = y \ ? \ 0 : 1))$
   c. $(y = 0 \ ? \ f : g)(17)$
   d. $b[0][1]$ (presumably b would be an array of arrays)
   e. $b[0,1]$ (where b would be a 2-dimensional array of integers)
2. Consider the state σ_2 described graphically below. Write a definition for σ_2 as a standard state predicate.

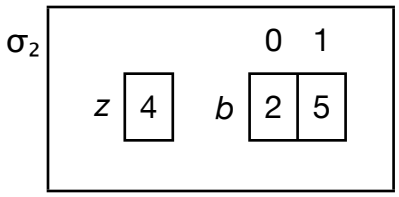


3. Let $\sigma_3 \equiv z = 4 \wedge b[0] = 2 \wedge b[1] = 5$.
   a. Abbreviate this using tuple notation for the value of b.
   b. Write out the graphical representation of the σ_3 (a diagram as in question 2).

4. Let $\varphi \equiv x{=}y{+}1 \wedge y{=}z^2{-}3 \wedge z{=}6$. Write out the definition of a standard state σ_4 equivalent to φ.
5. ~~Let $\sigma_5 \equiv x = 0 \wedge y = 5 \wedge b = (2, 4, 6, 8)$.~~
   a. ~~Does $\sigma_5 \vDash x + y < b[b[0]]$?~~
   b. ~~Does $\sigma_5 \vDash \forall\, i : 0 \le i < size(b) \rightarrow x < b[i]$?~~
   c. ~~Does $\sigma_5 \vDash \exists\, i : 0 \le i < size(b) \wedge b[i] > y$?~~
   d. ~~Is $0 \le i < size(b) \rightarrow b[i] < 9$ valid?~~
   e. ~~Is $b[i] < 9$ valid?~~

D. Solutions

1. legal; legal; illegal (expressions can't yield functions); illegal (no arrays of arrays); legal.
2. Abbreviated: $\sigma_2 \equiv x = 12 \wedge y = T \wedge b = (2, 4, 3, 8)$
   Not abbreviated: $\sigma_2 \equiv x = 12 \wedge y = T \wedge b[0] = 2 \wedge b[1] = 4 \wedge b[2] = 3 \wedge b[3] = 8$
3. $\sigma_3 \equiv z = 4 \wedge b = (2, 5)$



4. {x = 34 , y = 33 , z = 6}

