

Lecture 5: Primitive Types and Classes

CS 51G
Spring 2018
Kim Bruce

Announcements

- Discuss Exercise 4.10.6
- Must bring a design for part 1 of the program to beginning of lab.
 - Download form from lab description web page
 - Mentors and I will take a look and give feedback.
 - How can you efficiently check for correct basket?
- Lab will start with 15 minute quiz.

Boolean

- Boolean type has values true, false
 - var dragging: Boolean := true
- Operations: &&, || for “and” and “or”
 - “!p” returns opposite of p
- Relations ==, !=, <, <=, >, and >= all return values of type Boolean
- Use in “if” statements
- Boolean variables help us remember what happened earlier!

Examples

- Redo BetterBasketball
 - if (dragging && hoop.contains(point)) then {...}
- Review Pong bounds on paddle in onMouseMove

Types in Grace

- Predetermined types:
 - Number, Point, String, Graphic2D, Line, Boolean, Color, Done
- Will be creating own types soon
- Provide guarantee of operations available
- Provide error checking, documentation
- See specifications of types in documentation web page, esp. Number, Point

Numeric Operations

- +, -, *, /, % (modulo - remainder)
- unary minus (also written “-“)
- Precedence of operators:
 - Do *, / before +, -
 - Within groups go from left to right unless parens
 - So $2 + 6 * 5 / 2 - 12/4 = (2 + ((6 * 5) / 2)) - (12 / 4)$

Convert to integer

- Methods
 - truncated — throws away decimal: $(3.75).truncated = 3$
 - rounded — rounds to nearest int: $(3.75).rounded = 4$

Objectdraw Types

- Graphic2D, Line, etc. in objectdraw documentation
- Framed and filled rects and ovals have type Graphic2D
- See documentation for Application and GraphicApplication

Time in Grace

- Module `sys` provides method `elapsedTime`
 - returns number of seconds since an arbitrary start time
 - Subtract later one from earlier to find actual elapsed time.
- Examples:
 - <http://www.cs.pomona.edu/classes/cs05IG/demos/ClickTimer/ClickTimer.grace>
 - <http://www.cs.pomona.edu/classes/cs05IG/demos/MorseCode/MorseCode.grace>

Defining New Objects

- What is an object?
 - Consists of defs, vars, and methods
 - Methods provide operations that others can request
- Can define an object directly
 - <http://www.cs.pomona.edu/classes/cs051G/demos/ClassyBasketballObject/ClassyBasketballObject.grace>
- Can also define class to generate them:
 - <http://www.cs.pomona.edu/classes/cs051G/demos/ClassyBasketball/ClassyBasketball.grace>

Questions?