

## Objectives

- Picasso Design/Parsing

Nov 30, 2009

Sprenkle - CS209

1

## Project Deliverables Timeline

Deliverable	Who	Weight	Due Date
Preparation	Individual	8%	12/2
Preliminary Implementation	Team	37%	12/7
Final Implementation	Team	40%	You decide → latest 12/18
Analysis	Individual	15%	12/18

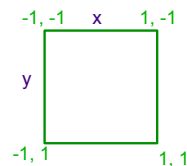
Nov 30, 2009

Sprenkle - CS209

2

## Picasso Project Overview

- Goal: Generate images from expressions
- Every pixel gets assigned a color, computed from its x and y coordinate and the given expression
- Colors are RGB values
  - Range  $[-1, 1]$
  - Black is  $[-1, -1, -1]$
  - Red is  $[1, -1, -1]$
  - Yellow is  $[1, 1, -1]$



Nov 30, 2009

Sprenkle - CS209

3

## Examples



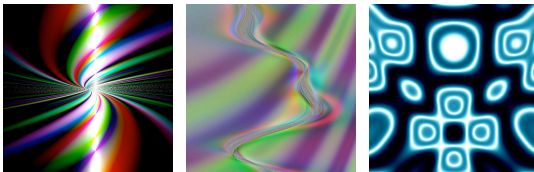
Nov 30, 2009

Sprenkle - CS209

4

## Specification

- User can enter expressions
  - Interactively or from file
  - Language is defined in specification
- Lots of possible extensions



Nov 30, 2009

Sprenkle - CS209

5

## Programming Language Syntax

- What does an identifier look like in Java?
- What does an assignment statement look like in Java?
- What can be on the left hand side?
- What can be on the right hand side?
- What does a multiplication look like?
- How do we evaluate arithmetic expressions?

Nov 30, 2009

Sprenkle - CS209

6

## Programming Language Design

- Must be unambiguous
  - Programming Language defines a syntax and semantics
- Interpreting programming languages
  - Parse program into tokens
    - Example:  $x = 4 * 3$ ;  $\rightarrow$

`<id> <assignment> <num> <mult> <num> <endofstmt>`

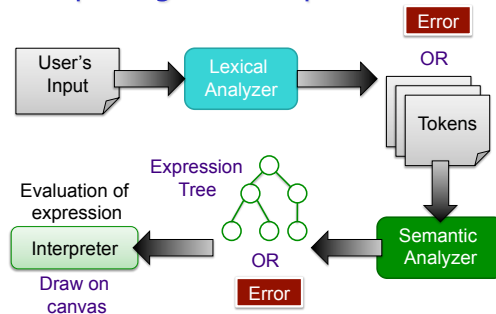
- Verify that tokens are in a valid form
- Generate executable code

Nov 30, 2009

Sprenkle - CS209

7

## Interpreting User's Input

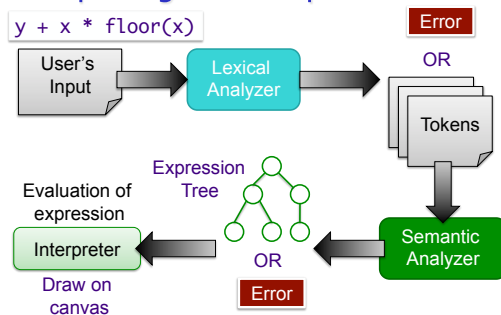


Nov 30, 2009

Sprenkle - CS209

8

## Interpreting User's Input



Nov 30, 2009

Sprenkle - CS209

9

## What We Need to Do/Represent

- Lexical Analysis
- Semantic Analysis
- Evaluation

Nov 30, 2009

Sprenkle - CS209

10

## What We Need to Do/Represent

- Lexical Analysis
  - Recognize/create tokens
  - Report errors in creating tokens
- Semantic Analysis
  - Convert infix tokens into postfix
    - Report errors
  - Parse tokens into *expressions*
    - Report errors
- Evaluation
  - Evaluate expressions with respect to  $x$  and  $y$

Nov 30, 2009

Sprenkle - CS209

11

## Picasso Starting Code

- Import an existing project from
  - `/home/courses/cs209/handouts/picasso.tar`
- Code has been updated since previous version
- Some errors in giving testing code. Most will be fixed by end of today.

Nov 30, 2009

Sprenkle - CS209

12

## Understanding the Code

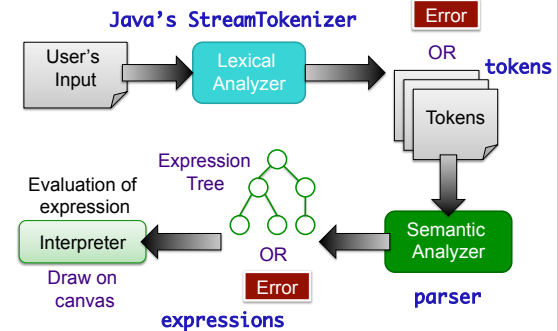
- What are the different parts of the code?

Nov 30, 2009

Sprenkle - CS209

13

## Interpreting User's Input



Nov 30, 2009

Sprenkle - CS209

14

## Understanding the Code

- Lexical analysis
  - `picasso.parser.Tokenizer`
  - `picasso.parser.tokens.TokenFactory`
  - Output: `picasso.parser.tokens.*`

FloorToken

Nov 30, 2009

Sprenkle - CS209

15

## Understanding the Code

- Semantic analysis
  - `picasso.parser.ExpressionTreeGenerator`
  - `picasso.parser.SemanticAnalyzer`
  - `picasso.parser.*Analyzer`
  - Output: `picasso.parser.language.expressions.*`

FloorAnalyzer

Nov 30, 2009

Sprenkle - CS209

16

## Understanding the Code

- Evaluation
  - Base class: `picasso.parser.language.ExpressionTreeNode`
  - Output: `RGBColor`

Floor

Nov 30, 2009

Sprenkle - CS209

17

## Practice Adding Functions

- Create a token for the sine function
  - Same prefix as new function, e.g., `SinToken.java`
  - Needs to be added to `functions.conf`
- Create a semantic analyzer for the function with same prefix as function, e.g., `SinAnalyzer.java`
  - `Analyzer` class (presumably implementing `SemanticAnalyzerInterface`) returns an instance of `ExpressionTreeNode`
- Create an `ExpressionTreeNode` for function `Sine.java`

Nov 30, 2009

Sprenkle - CS209

18

## What Do You Think You'll Need To Do About Binary Operators?

Nov 30, 2009

Sprenkle - CS209

19

## For Wednesday: Project Preparation

- Read over the Picasso (Final Project) specifications again
- 1<sup>st</sup> deliverable is a text document that answers
  - What needs to be completed?
  - What is your plan for completing those tasks?
  - What tasks are you most interested in working on?
  - ....
- Wednesday
  - Discuss your plans, questions
  - Discuss tools to help collaboration

Nov 30, 2009

Sprenkle - CS209

20