Zoë Gordin, Eleanor Murguia, Nadia Saleh
Motivation

● Create a general purpose programming language that is specialized for graphics
  ○ Patterns, Fractals
● Fractals are hard to draw by hand, why shouldn’t the computer draw them for us?!
● Drawing shapes is fun!
<table>
<thead>
<tr>
<th>Date</th>
<th>Milestone</th>
</tr>
</thead>
<tbody>
<tr>
<td>September 19</td>
<td>Proposal Due</td>
</tr>
<tr>
<td>October 1</td>
<td>Revised Proposal</td>
</tr>
<tr>
<td>October 12</td>
<td>First commit in Project Repo</td>
</tr>
<tr>
<td>October 15</td>
<td>LRM, Parser</td>
</tr>
<tr>
<td>October 22</td>
<td>AST</td>
</tr>
<tr>
<td>October 30</td>
<td>SAST, Started Semant</td>
</tr>
<tr>
<td>November 6</td>
<td>Started Codegen</td>
</tr>
<tr>
<td>November 14</td>
<td>Hello World</td>
</tr>
<tr>
<td>December 3</td>
<td>External Library Linked</td>
</tr>
<tr>
<td>December 18</td>
<td>Tuples, Arrays Completed</td>
</tr>
<tr>
<td>December 19</td>
<td>Final Report Completed</td>
</tr>
</tbody>
</table>
Features

- Modulo
  - Useful for patterns

- Arrays
  - Mutable
  - Useful for iteratively creating similar shapes

- Tuples
  - Immutable
  - Useful for coordinates

- Strings
  - Useful for debugging and logging messages

- SIGIL
  - Circles, Rectangles, Triangles, Lines, Points
System Architecture

- source code
- Scanner
  - tokens
- Parser
  - syntax
- AST
  - AST
- Semant
  - SAST
  - LLVM
- Executable
  - gcc
- Codegen
- SAST
- libsigil.so
Testing

Methodology

- test.sh testfile.zen
  - Runs testfile.zen, writes to testfile.out
- Testall.sh
  - Runs all tests.zen, compares output to expected test.out or test.err

Challenges

- Sigil windows can’t be saved and compared
- Can only check that graphical programs execute, not that they produce the desired output
Development Challenges

- **Graphical Libraries**
  - Picking the best one
  - Docker cannot support!!
  - Testing

- **Library Linking**
  - Adding multiple built-in library functions with different arguments

- **Aggregate Types**
  - Creating and accessing them in codegen
### Contributions and Lessons Learned

<table>
<thead>
<tr>
<th>Name</th>
<th>Contribution</th>
<th>Lessons Learned</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zoë Gordin</td>
<td>Arrays, Library Linking</td>
<td>Adding to an existing codebase is <em>hard</em>, the power of Git, TAs are your friends!</td>
</tr>
<tr>
<td>Eleanor Murguia</td>
<td>Tuples, Demo</td>
<td>Fixes are a less complicated than you think they will be (thanks OCaml!)</td>
</tr>
<tr>
<td>Nadia Saleh</td>
<td>Tests</td>
<td>Every change will probably break an existing test. Run them all, often!</td>
</tr>
</tbody>
</table>
Demo!