Studio model: “contracts”

What’s your take?

What does the studio model mean to you?

How do we interpret the Honor Code?

How does the course design affect assessment / grades?

How will you take advantage?

1. Suggest at least two good practices
2. Identify at least two warning signs
3. Ask at least two “unknowns”
   Conundrums / gray areas
   Questions / confusions
What is DSL?

Today’s themes: tradeoffs and gray areas
What is a Programming Language?

- Can you name some things that are programming languages?
- Can you name some things that are not programming languages?
- What must every programming language (not) have?
- Write down a succinct definition of programming language.
Today’s class: discussion questions

1. Mini groups of three (can’t include your critique partner)
   • Discussion questions on handouts
   • Take notes
   • Take the full time

2. Enumerate the insights / questions

3. Full-class discussion
   • We’ll revisit the questions, see where we (dis)agree

4. Info: critiques & schedule

5. Is this a DSL? (If there’s time)
What is a programming language?
DSL case studies
What is a DSL?
Critique the evolution of Context Free
Enumerate the insights!
Discussion Rules

• Everyone should participate equally
• Include everyone
• Yes, and… (build on discussion)
• (Prof.) Ben usually talks last
• Give your attention to the person who is speaking
• …
What is a programming language?
void StringStack::push(const string pushee) {
    assert(count_ < MAX_ELEMENTS);
    store_[count_++] = pushee;
}
for person in people:
    print person.name
Our active <a href="/research/">Research Program</a> involves approximately 50 students in research, funded by grants and gifts from the National Science Foundation, Howard Hughes Medical Institute, and the Rose-Hills and Baker Foundations.
Genetics
What is a DSL?
What is a *Domain-Specific Language (DSL)*?

**Touchstones**

- What should be easy in this language?
- What should be difficult?
- What should be impossible?
Critique the evolution of Context Free
Critiques

- Give yourself a budget of ~90 minutes
- Quality, not necessarily quantity
  Engage. Be intentional and **specific**. Small bite, long chew!
- Some useful critique prompts
  - Prompts from the assignment
  - If someone asks a question, attempt an answer
  - If someone asks for feedback, give it
  - “I hadn’t thought of that! Can I steal it?”
  - “You could also…”
  - “I found…”
  - “I could use this for …”
<table>
<thead>
<tr>
<th>Language design</th>
<th>Prep</th>
<th>Tools: Internal DSLs</th>
<th>Tools: External DSLs</th>
<th>Projects</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Monday</strong></td>
<td><strong>Wednesday</strong></td>
<td><strong>Monday</strong></td>
<td><strong>Wednesday</strong></td>
<td><strong>Monday</strong></td>
</tr>
<tr>
<td>09 - 08</td>
<td>(no class)</td>
<td>09 - 04</td>
<td>What is a DSL?</td>
<td>09 - 04</td>
</tr>
<tr>
<td>09 - 09</td>
<td>What is a DSL?</td>
<td>09 - 11</td>
<td>Mini-tab: language design</td>
<td>09 - 18</td>
</tr>
<tr>
<td>09 - 16</td>
<td>Projects</td>
<td>09 - 25</td>
<td>Scala intro</td>
<td>09 - 25</td>
</tr>
<tr>
<td>09 - 23</td>
<td>Functional Scala</td>
<td>10 - 03</td>
<td>Lab: Internal control flow</td>
<td>10 - 03</td>
</tr>
<tr>
<td>10 - 07</td>
<td>Extensibility: The Expression Problem</td>
<td>10 - 09</td>
<td>Language architecture &amp; parser combinators</td>
<td>10 - 09</td>
</tr>
<tr>
<td>10 - 14</td>
<td>Models of computation</td>
<td>10 - 16</td>
<td>Lessons and plans</td>
<td>10 - 16</td>
</tr>
<tr>
<td>10 - 21</td>
<td>Fall Break (no class)</td>
<td>10 - 23</td>
<td>studio</td>
<td>10 - 23</td>
</tr>
<tr>
<td>10 - 28</td>
<td>Language features, part 1</td>
<td>10 - 30</td>
<td>studio</td>
<td>10 - 30</td>
</tr>
<tr>
<td>11 - 04</td>
<td>Language features, part 2</td>
<td>11 - 05</td>
<td>studio</td>
<td>11 - 05</td>
</tr>
<tr>
<td>11 - 11</td>
<td>studio</td>
<td>11 - 13</td>
<td>Language workbenches</td>
<td>11 - 13</td>
</tr>
<tr>
<td>11 - 18</td>
<td>studio</td>
<td>11 - 20</td>
<td>studio</td>
<td>11 - 20</td>
</tr>
<tr>
<td>11 - 25</td>
<td>Logos</td>
<td>11 - 27</td>
<td>Thanksgiving (no class)</td>
<td>11 - 27</td>
</tr>
<tr>
<td>12 - 02</td>
<td>studio</td>
<td>12 - 04</td>
<td>studio</td>
<td>12 - 04</td>
</tr>
<tr>
<td>12 - 09</td>
<td>Presentations</td>
<td>13 - 11</td>
<td>Presentations</td>
<td>13 - 11</td>
</tr>
</tbody>
</table>
Before Wednesday

• Write your contracts

• Assignment 1
  critique due Tuesday

• Write up today’s class
  And everyone can / should contribute.

• Restore furniture

• Bring your computer on Wednesday (if you want)
  It should have Python on it • We’ll also have computers here