Increasing Engagement in Automata Theory with JFLAP

Susan H. Rodger
Duke University
rodger@cs.duke.edu

SIGCSE 2009
March 7, 2009

Supported by NSF Grant DUE 0442513

Outline

• Introduction and Overview of JFLAP
• JFLAP Two-year Study and Results
• New Items in JFLAP
  – User Control Parser
  – CYK Parser
  – Expanding Pumping Lemma
  – Turing machine to Unrestricted Grammar
  – Others
• Conclusions and Future Work

Co-Authors

• Evaluators - NCSU
  – Eric Wiebe
  – Kareem Omar
• Undergraduates - Duke
  – Kyung Min Lee
  – Chris Morgan
  – Jonathan Su

Formal Languages and Automata Theory

• Traditionally taught
  – Pencil and paper exercises
  – No immediate feedback
• Different
  – More mathematical than most CS courses
  – Less hands-on than most CS courses
  – Programming is in most of their CS courses, not here
Why Develop Tools for Automata?

<table>
<thead>
<tr>
<th>Textual</th>
<th>( {(q_0, a, q_0), (q_0, b, q_0), (q_0, \varepsilon, q_1)} )</th>
</tr>
</thead>
</table>
| Tabular | \begin{tabular}{|c|c|}
<table>
<thead>
<tr>
<th>a</th>
<th>b</th>
</tr>
</thead>
<tbody>
<tr>
<td>( q_0 )</td>
<td>( q_2 )</td>
</tr>
<tr>
<td>( q_1 )</td>
<td>( q_2 )</td>
</tr>
<tr>
<td>Interactive</td>
<td></td>
</tr>
</tbody>
</table>

Overview of JFLAP

- **Java Formal Languages and Automata Package**
- Instructional tool to learn concepts of Formal Languages and Automata Theory
- Topics:
  - Regular Languages
  - Context-Free Languages
  - Recursively Enumerable Languages
  - Lsystems
- With JFLAP your creations come to life!

Thanks to Students - Worked on JFLAP and Automata Theory Tools

- NPDA - 1990, C++, Dan Caugherty
- JFLAP - 1996-1999, Java version
  Eric Gramond, Ted Hung, Magda and Octavian Procopiuc
- Pâté, JeLLRap, Lsys
  Anna Bilska, Jason Salemme, Lenore Ramm, Alex Karweit, Robyn Geer
- JFLAP 4.0 – 2003, Thomas Finley, Ryan Cavalcante
- JFLAP 6.0 – 2005-2008 Stephen Reading, Bart Bressler, Jinghui Lim, Chris Morgan, Jason Lee, Jonathan Su

JFLAP – Regular Languages

- Create
  - DFA and NFA
  - Moore and Mealy
  - regular grammar
  - regular expression
- Conversions
  - NFA to DFA to minimal DFA
  - NFA \( \leftrightarrow \) regular expression
  - NFA \( \leftrightarrow \) regular grammar
JFLAP – Regular languages (more)

- Simulate DFA and NFA
  - Step with Closure or Step by State
  - Fast Run
  - Multiple Run
- Combine two DFA
- Compare Equivalence
- Brute Force Parser
- Pumping Lemma

JFLAP – Context-free Languages

- Create
  - Nondeterministic PDA
  - Context-free grammar
  - Pumping Lemma
- Transform
  - PDA → CFG
  - CFG → PDA (LL & SLR parser)
  - CFG → CNF
  - CFG → Parse table (LL and SLR)
  - CFG → Brute Force Parser

JFLAP – Recursively Enumerable Languages

- Create
  - Turing Machine (1-Tape)
  - Turing Machine (multi-tape)
  - Building Blocks
  - Unrestricted grammar
- Parsing
  - Unrestricted grammar with brute force parser

JFLAP - L-Systems

- This L-System renders as a tree that grows larger with each successive derivation step.
Students love L-Systems

Other Tools for Automata

• Turing’s World (Barwise and Etchemendy)
• Deus Ex Machina (Taylor and Savoiu)
• Theory of Computing Hypertextbook (Ross)
• Many others
  – L-System tools
  – Compiler tools
  – Finite State machine tools

JFLAP Study

• Study of JFLAP’s effectiveness in learning
  – Two year study
  – Fourteen Faculty Adopters
  – Two 2-day faculty Adopter Workshops – June 2005, June 2006
  – Collect data 2005-06 and 2006-07 Academic years
  – Pretest/Posttest
  – Interviews
  – Team of three evaluators
    • Eric Weibe – Education
    • Rocky Ross – Computer Science Theory
    • Joe Bergin – Computer Science Tools

Fourteen Faculty Adopter Participants

- small, large
- public, private
- includes minority institutions

• Duke
• UNC-Chapel Hill
• Emory
• Winston-Salem State University
• United States Naval Academy
• Rensselaer Polytechnic Institute
• UC Davis
• Virginia State University
• Norfolk State University
• University of Houston
• Fayetteville State University
• University of Richmond
• San Jose State University
• Rochester Institute of Technology
We hoped to show with this learning approach...

- Students gain a better and deeper understanding of FLA
- Students are happier and more confident in learning FLA
- Students are more interested in using the tools on their own
- Instructors can easily use the tools in class
- Instructors can easily grade electronic submissions

Goals of the JFLAP Study - Formal Languages and Automata (FLA)

- Present FLA in a visual and interactive manner in addition to the more traditional approach
  - Integrated
- Present Applications of FLA
- Provide a tool for allowing students to explore FLA in a computational manner
- Provide Materials for instructors to integrate this approach in their courses

Running a Study is hard!

- Hit by the drop in enrollments in after dot-com burst
- IRBs are different process at every institution
  - One page writeup ok’d (simplest)
  - Full medical IRB (many pages)
- One institution shut down all IRB research projects – we could not use data already collected.
- One University - Control Group – different times means different types of students, different professors.
- Some faculty came to workshop and did not follow through
- There were also some fantastic faculty!

Year One Instructor Interviews

- Used JFLAP in their courses
  - Primary use in class – demonstrations
  - Some used it to generate the graphics for their lecture
  - Extensive use – homeworks – includes electronic submission
  - One used it in office hours
### Year One – Software Implementation

<table>
<thead>
<tr>
<th>Question</th>
<th>YES/NO</th>
<th>No.</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Did you use JFLAP software to study for inclass exams?</td>
<td>YES</td>
<td>20</td>
<td>55%</td>
</tr>
<tr>
<td></td>
<td>NO</td>
<td>16</td>
<td>45%</td>
</tr>
<tr>
<td>Did you feel you had time to learn how to use the JFLAP software?</td>
<td>YES</td>
<td>33</td>
<td>94%</td>
</tr>
<tr>
<td></td>
<td>NO</td>
<td>2</td>
<td>6%</td>
</tr>
<tr>
<td>Did you feel that using the software took time away from other study activities?</td>
<td>YES</td>
<td>3</td>
<td>8%</td>
</tr>
<tr>
<td></td>
<td>NO</td>
<td>33</td>
<td>92%</td>
</tr>
<tr>
<td>Did the time and effort it took to use JFLAP help you get a better grade in the course?</td>
<td>YES</td>
<td>23</td>
<td>64%</td>
</tr>
<tr>
<td></td>
<td>NO</td>
<td>13</td>
<td>36%</td>
</tr>
<tr>
<td>Was it easier to use JFLAP software or was it easier to draw it out by hand?</td>
<td>software</td>
<td>30</td>
<td>83%</td>
</tr>
<tr>
<td></td>
<td>by hand</td>
<td>6</td>
<td>17%</td>
</tr>
<tr>
<td>Did you feel you would have done as well in the course if you had not used JFLAP?</td>
<td>YES</td>
<td>18</td>
<td>50%</td>
</tr>
<tr>
<td></td>
<td>NO</td>
<td>13</td>
<td>36%</td>
</tr>
<tr>
<td></td>
<td>NA</td>
<td>5</td>
<td>14%</td>
</tr>
</tbody>
</table>

### Years 1 and 2: Usability Survey

<table>
<thead>
<tr>
<th>Question</th>
<th>Very Easy</th>
<th>Easy</th>
<th>Neither</th>
<th>Difficult</th>
<th>Very Difficult</th>
</tr>
</thead>
<tbody>
<tr>
<td>How easy was it to use the drawing tool of JFLAP?</td>
<td>31%</td>
<td>48%</td>
<td>15%</td>
<td>5%</td>
<td>0%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>How easy was it to run the automata you designed in JFLAP?</td>
<td>Very easy</td>
<td>Easy</td>
<td>Neither</td>
<td>Difficult</td>
<td>Very Difficult</td>
</tr>
<tr>
<td>(134 respondents)</td>
<td>53%</td>
<td>47%</td>
<td>12%</td>
<td>6%</td>
<td>2%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>How easy was it to interpret results from the test run in JFLAP?</td>
<td>Very Easy</td>
<td>Easy</td>
<td>Neither</td>
<td>Difficult</td>
<td>Very Difficult</td>
</tr>
<tr>
<td>(134 respondents)</td>
<td>23%</td>
<td>45%</td>
<td>19%</td>
<td>10%</td>
<td>5%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>What is your overall assessment of the JFLAP software?</td>
<td>Very Poor</td>
<td>Poor</td>
<td>Neither</td>
<td>Good</td>
<td>Very Good</td>
</tr>
<tr>
<td>(133 respondents)</td>
<td>2%</td>
<td>4%</td>
<td>4%</td>
<td>63%</td>
<td>20%</td>
</tr>
</tbody>
</table>

### Year 2 – Implementation Survey

<table>
<thead>
<tr>
<th>Question</th>
<th>Time</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>When preparing for exams what percentage of study time involved the use of JFLAP software?</td>
<td>0-20%</td>
<td>68%</td>
</tr>
<tr>
<td></td>
<td>21-40%</td>
<td>16%</td>
</tr>
<tr>
<td></td>
<td>41-60%</td>
<td>11%</td>
</tr>
<tr>
<td></td>
<td>61-80%</td>
<td>3%</td>
</tr>
<tr>
<td></td>
<td>81-100%</td>
<td>2%</td>
</tr>
<tr>
<td>How often did you use JFLAP to do additional practice problems?</td>
<td>Never</td>
<td>46%</td>
</tr>
<tr>
<td></td>
<td>Rarely</td>
<td>25%</td>
</tr>
<tr>
<td></td>
<td>Occasionally</td>
<td>21%</td>
</tr>
<tr>
<td></td>
<td>Often</td>
<td>4%</td>
</tr>
<tr>
<td></td>
<td>Very Often</td>
<td>4%</td>
</tr>
</tbody>
</table>

### Year 2 – Usability Survey

<table>
<thead>
<tr>
<th>Question</th>
<th>Strongly Agree</th>
<th>Agree</th>
<th>Neither</th>
<th>Disagree</th>
<th>Strongly Disagree</th>
</tr>
</thead>
<tbody>
<tr>
<td>Using JFLAP made the course more enjoyable for me.</td>
<td>12%</td>
<td>51%</td>
<td>25%</td>
<td>6%</td>
<td>4%</td>
</tr>
<tr>
<td>(98 responses)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Using JFLAP made me feel more engaged in the course.</td>
<td>Strongly Agree</td>
<td>Agree</td>
<td>Neither</td>
<td>Disagree</td>
<td>Strongly Disagree</td>
</tr>
<tr>
<td>(98 responses)</td>
<td>13%</td>
<td>59%</td>
<td>15%</td>
<td>9%</td>
<td>3%</td>
</tr>
<tr>
<td>Having access to JFLAP made learning course concepts ...</td>
<td>Much harder</td>
<td>Harder</td>
<td>Neither</td>
<td>Somewhat easier</td>
<td>Much easier</td>
</tr>
<tr>
<td>(97 responses)</td>
<td>1%</td>
<td>5%</td>
<td>26%</td>
<td>54%</td>
<td>14%</td>
</tr>
</tbody>
</table>
New Items in JFLAP

- User-Control Parser (UCP)
  - Allows for more exploration

UCP – select rule, select variable, Step

UCP – final parse tree

CYK Parser

- Cocke-Younger-Kasami (CYK)
- How it works in JFLAP
  - Internally, convert to CNF grammar
  - Then map back to original grammar
CYK – substantially faster than Brute Force Parser for many grammars

- Brute Force Parser
  8 secs

- CYK parser
  1 sec

Pumping Lemma Game – Linz Book

Turing Machine to Unrestricted Grammar

\[ ww^R \]
Other New Items

- Can automatically add a trap state
- Additional graph layout choices
- Can load an input string from a file
  – This is useful for universal Turing machine
- Online JFLAP Tutorial
- JFLAP wiki and listserv

Getting interaction into the automata theory course......

There are lots of ways to get interaction in this course...

Interaction in Class – Props Edible Turing Machine

- TM for $f(x) = 2x$ where $x$ is unary
- TM is not correct, can you fix it? Then eat it!
- States are blueberry muffins

Students Work in Groups to Solve Problems – With or W/out JFLAP

- Lecture some, then stop
- Students work on problem with JFLAP
- Bring students back together
Students building DFA with cookies and icing

JFLAP’s Use Around the World

- JFLAP web page has over 200,000 hits since 1996
- Google Search
  - JFLAP appears on over 9830 web pages
  - Note: search only public web pages
- JFLAP been downloaded in over 160 countries

Conclusions From Study

- Results of Study showed
  - All the faculty used JFLAP in their courses, mostly for homework, some in lecture
  - Students had a high opinion of JFLAP
  - Majority of students felt access to JFLAP
    - Made learning course concepts easier
    - Made them feel more engaged
    - Made the course more enjoyable
  - Over half the students used JFLAP to study for exams
  - Over half the student thought time and effort using JFLAP helped them get a better grade.

Resources/Questions?

JFLAP works well with Linz book

New CD supplement with JFLAP exercises to go with this book

JFLAP tutorial

JFLAP book