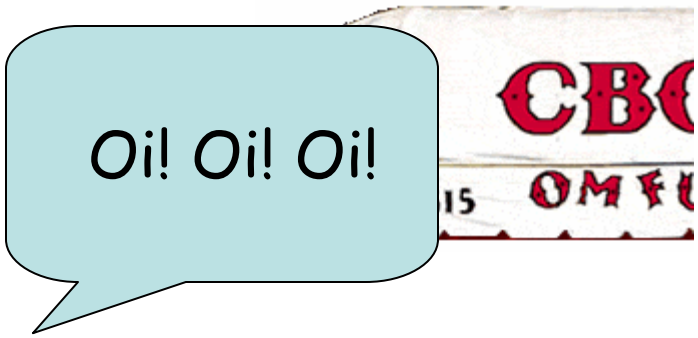
Chris Murphy

Ryan Overbeck

Lauren Wilcox

Joeng Kim

Introduction

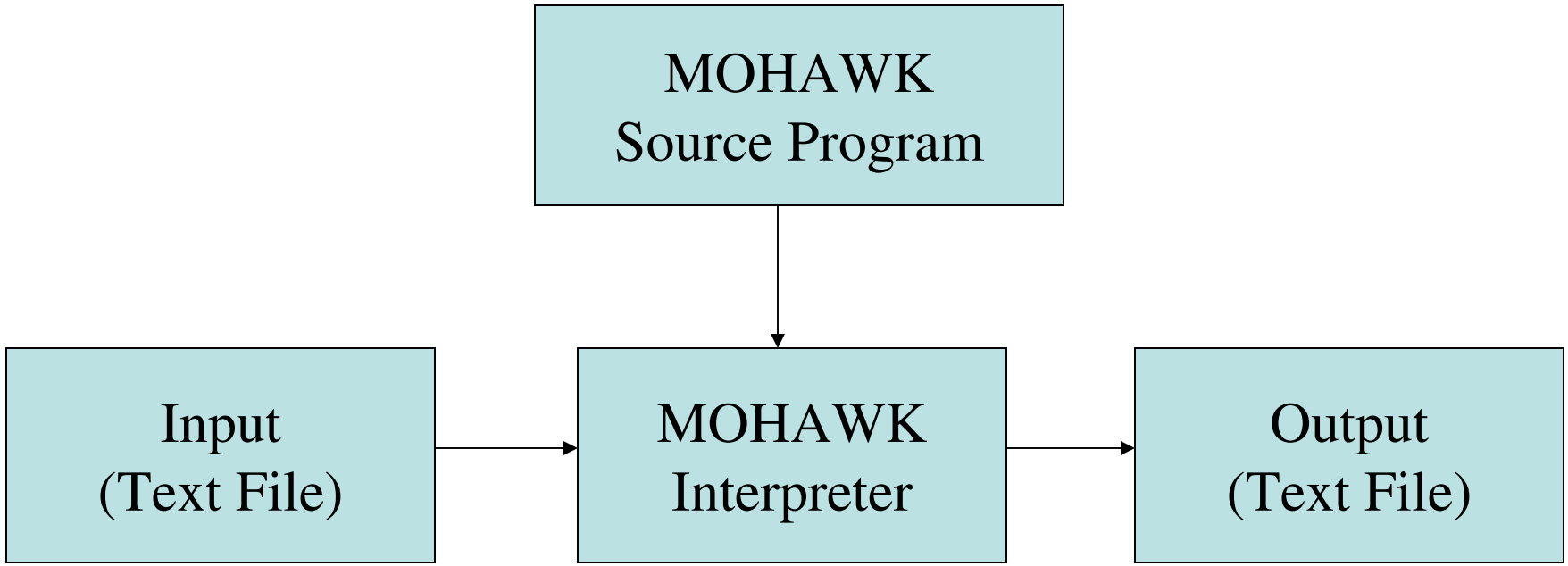


AWK

What is MOHAWK?

- A programming language for processing tabular data from text files
- Largely based on AWK

Runtime Architecture



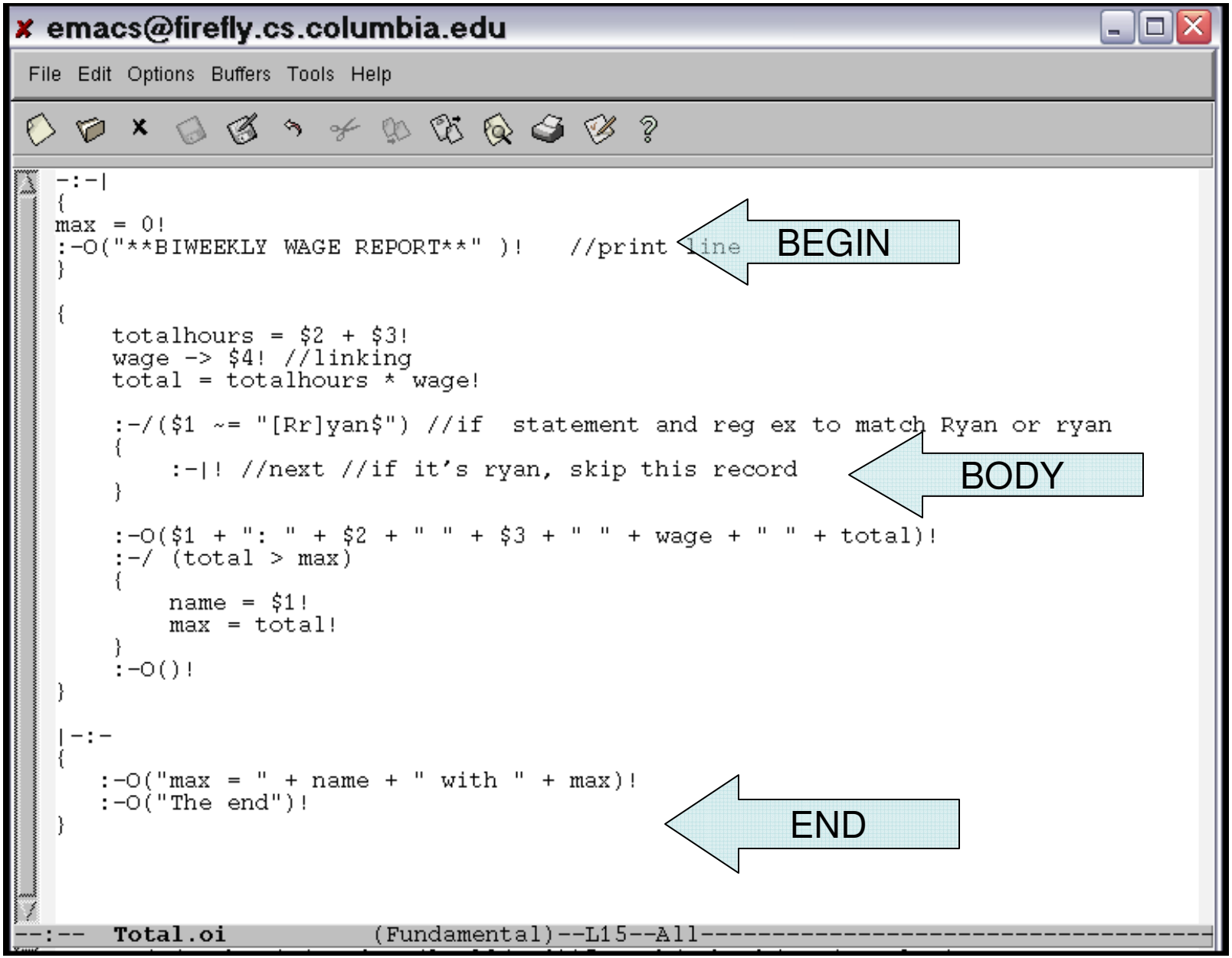
Mohawk Program

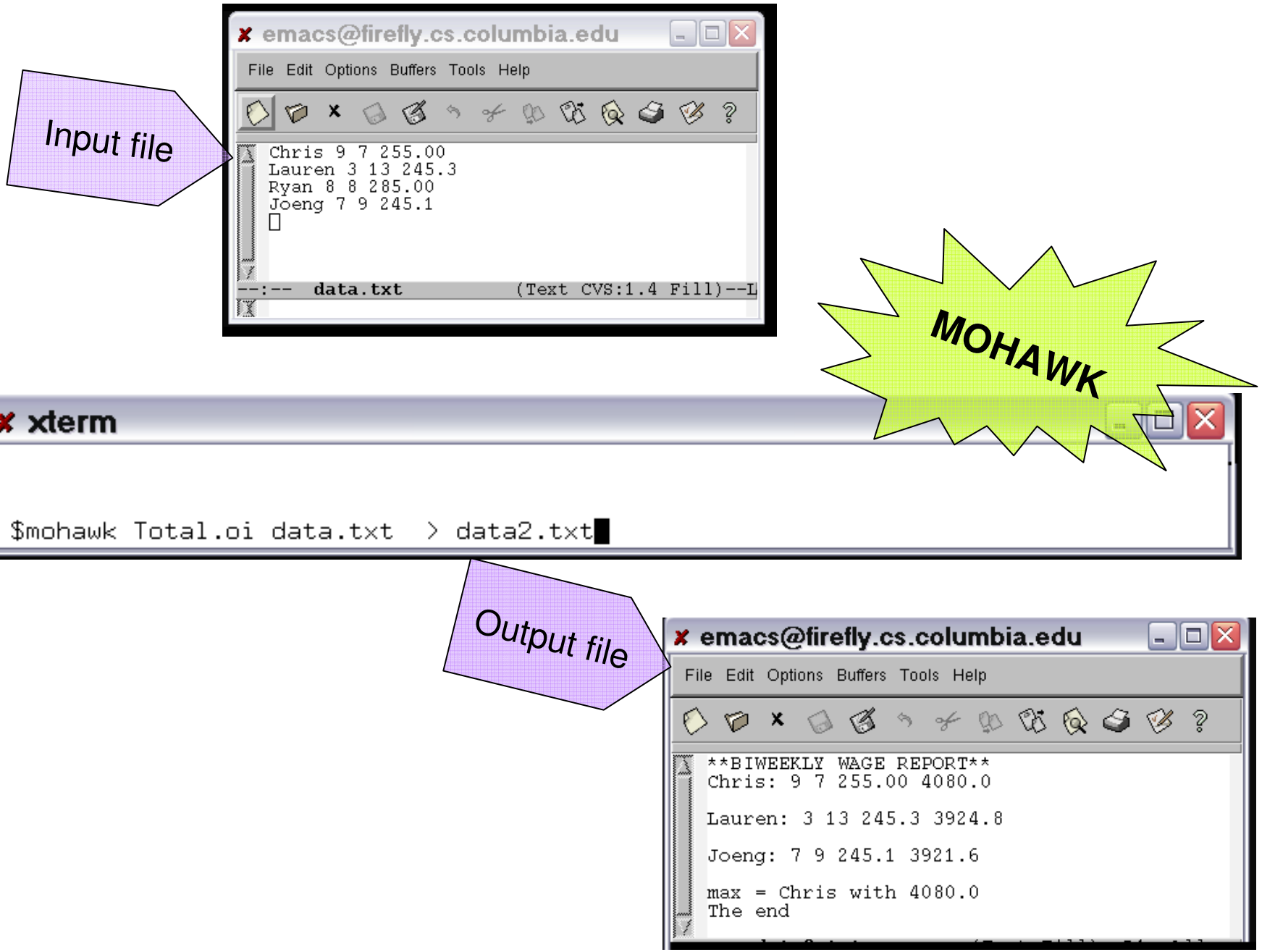
- A series of pattern/action statements
- Patterns are:
  - “-:-|” (Forward MOHAWK)
  - “|-:-” (Reverse MOHAWK)
  - A numeric, boolean, or string valued expression
  - Nothing.
- Actions are a series of:
  - Math, boolean, and string expressions
  - Conditional statements
  - Function calls
  - I/O Control

Distinguishing Features

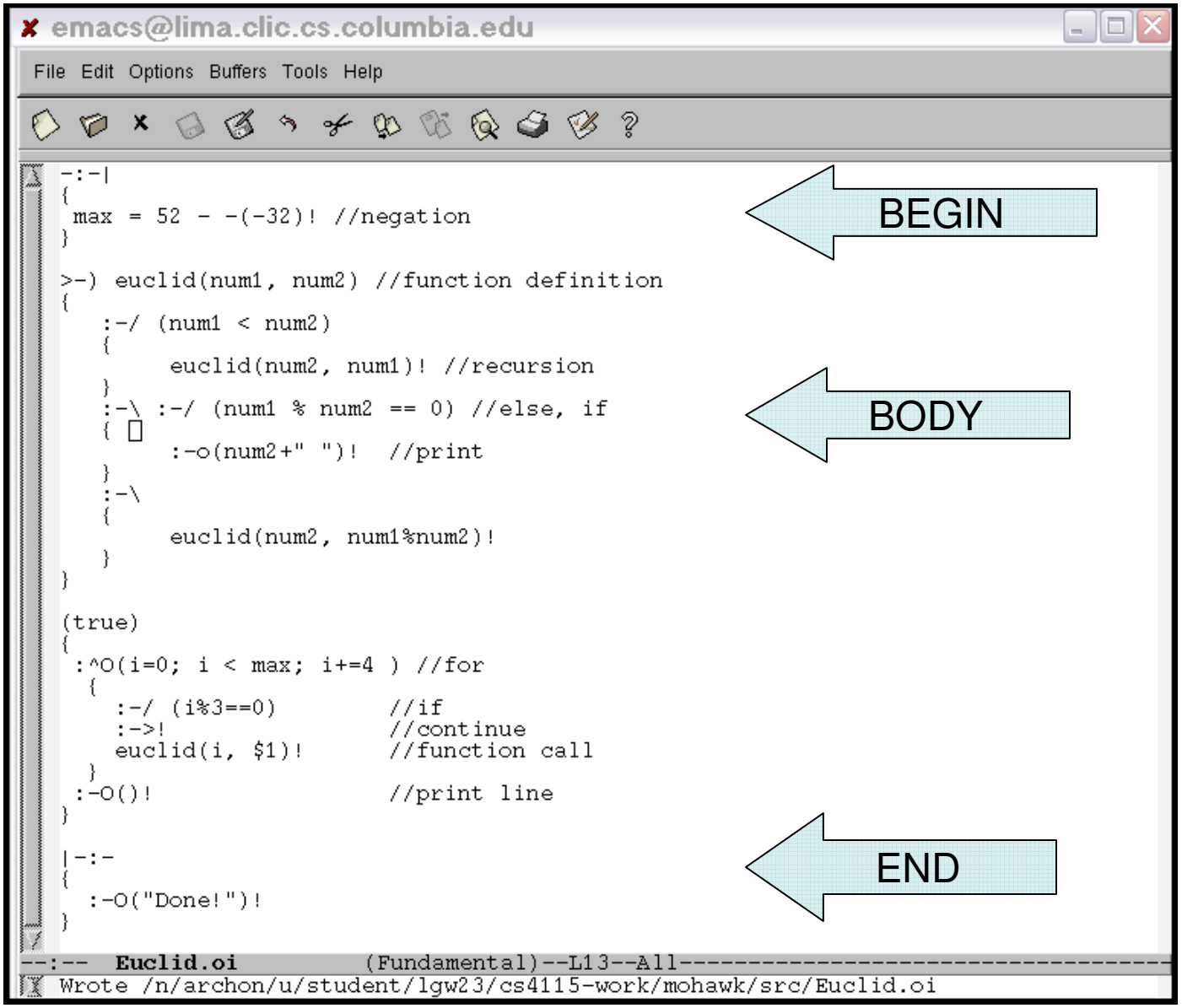
- Uses ‘!’ as a statement terminator
- No data types (like AWK)
- Java style comments
- Linking numbered fields with meaningful string identifiers
- New variable scoping rules
- Most keywords are IRC style smilies

Tutorial Introduction/Example

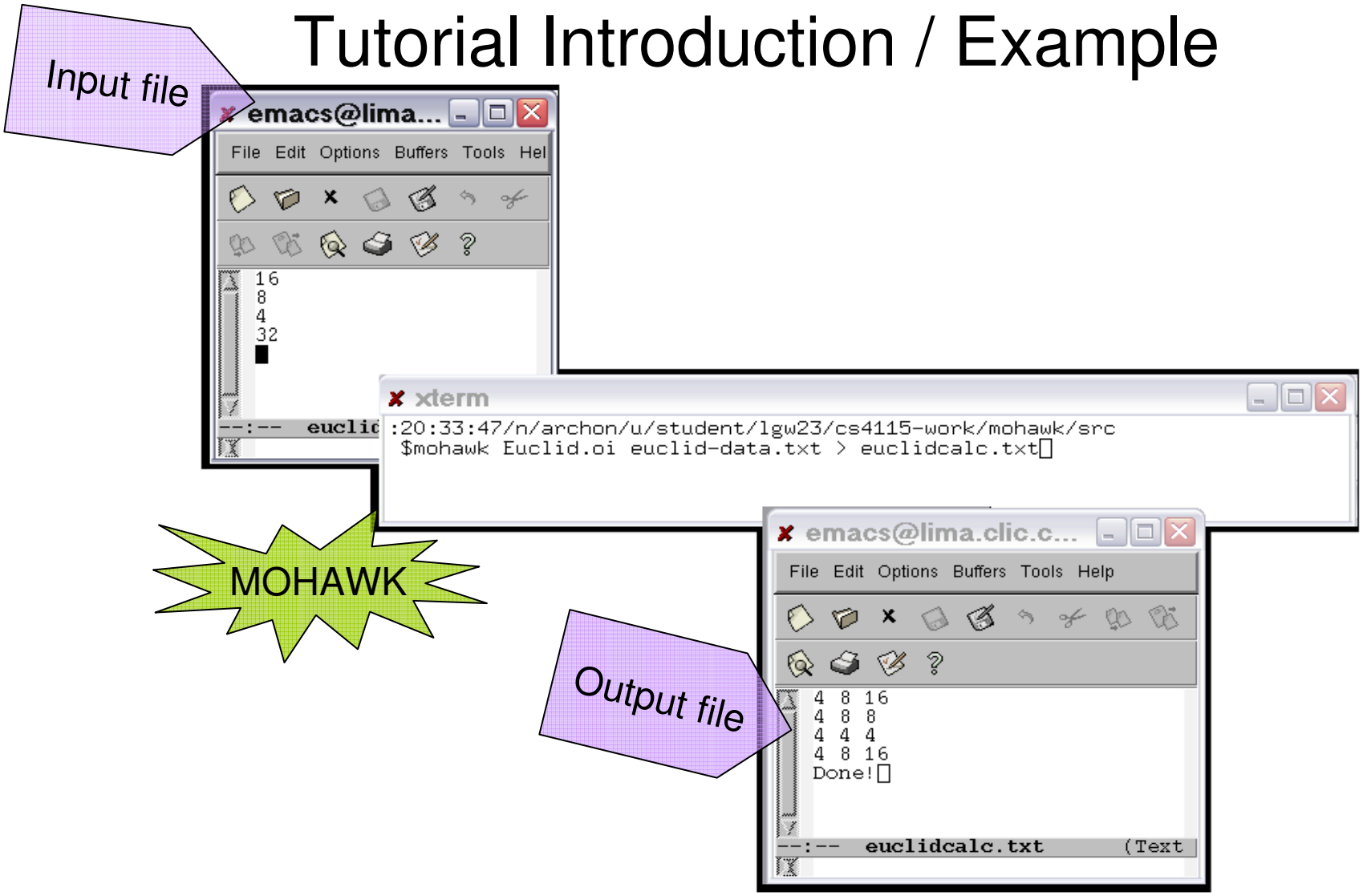



Input file
emacs@firefly.cs.columbia.edu
File Edit Options Buffers Tools Help
Chris 9 7 255.00
Lauren 3 13 245.3
Ryan 8 8 285.00
Joeng 7 9 245.1
--:-- data.txt (Text CVS:1.4 Fill)--L
MOHAWK
xterm
$mohawk Total.oi data.txt > data2.txt
Output file
emacs@firefly.cs.columbia.edu
File Edit Options Buffers Tools Help
BIWEEKLY WAGE REPORT
Chris: 9 7 255.00 4080.0
Lauren: 3 13 245.3 3924.8
Joeng: 7 9 245.1 3921.6
max = Chris with 4080.0
The end

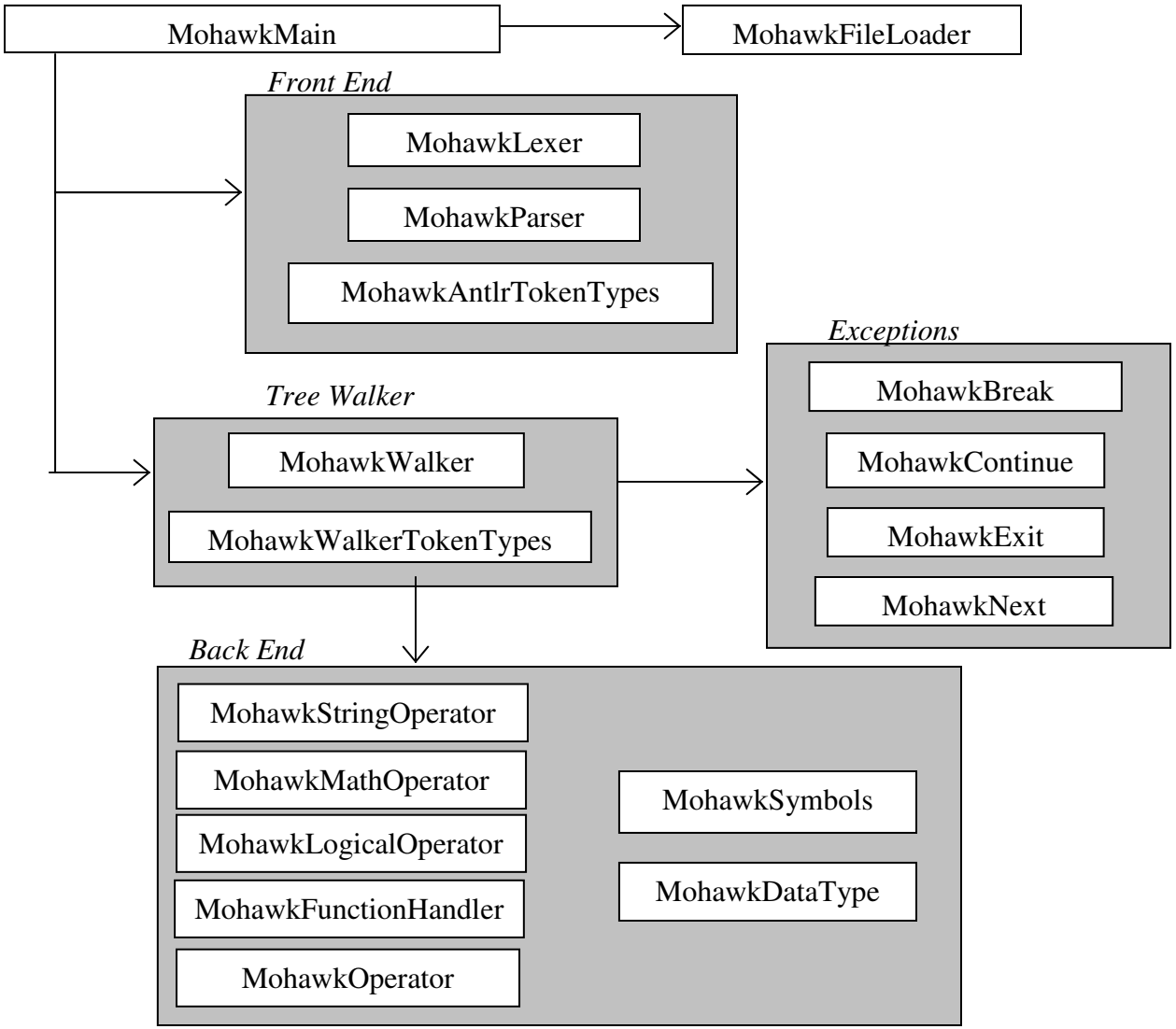
Tutorial Introduction / Example



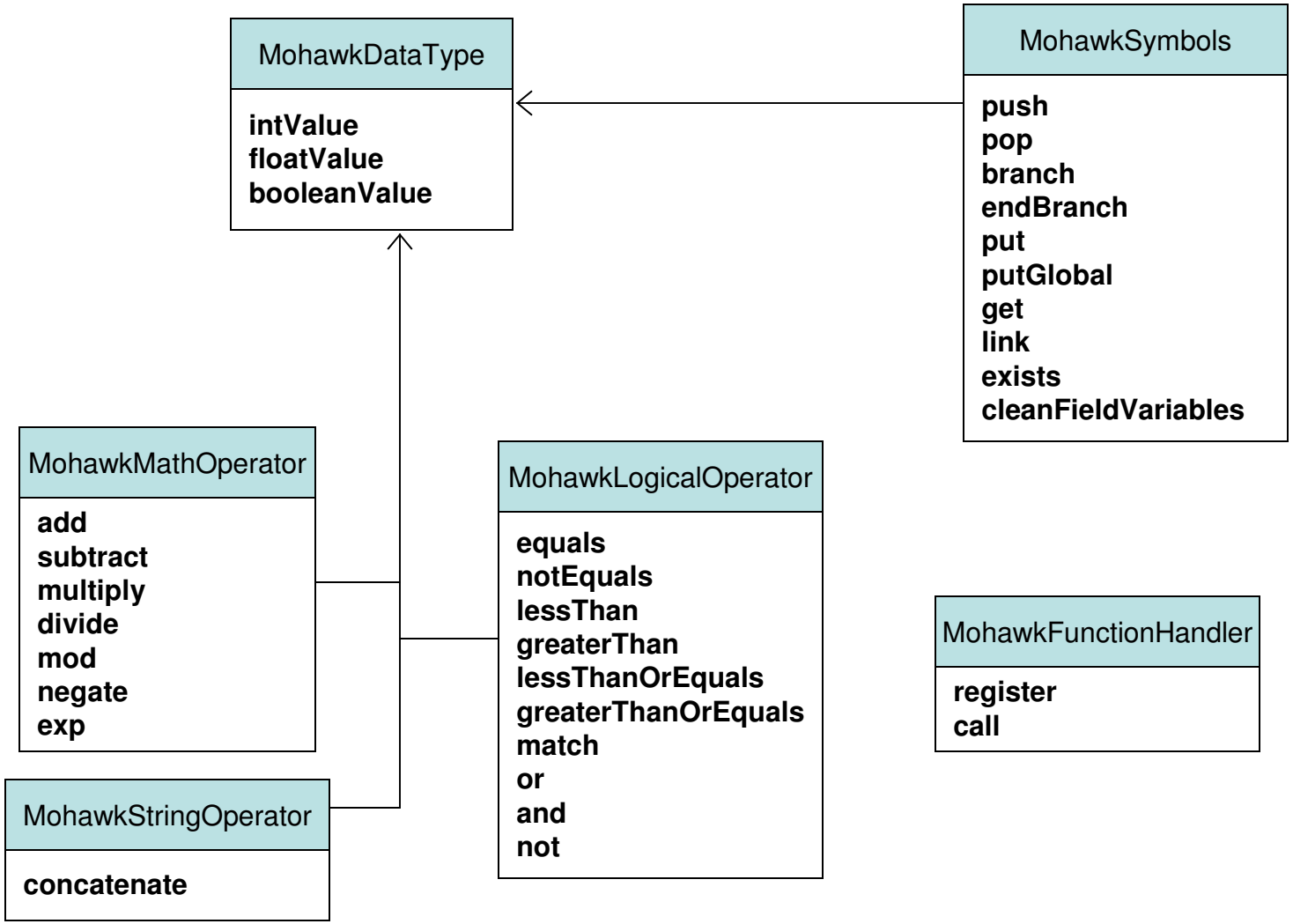
Tutorial Introduction / Example



MOHAWK Component Architecture

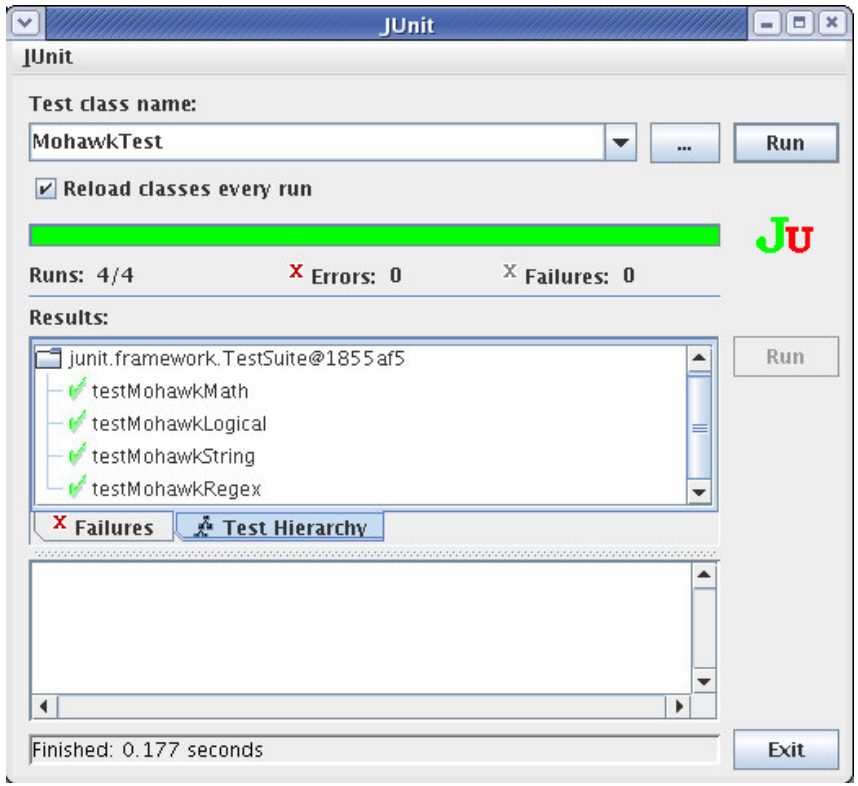


Back End Components



Testing

- Unit Testing
  - JUnit
  - Test operators
- Parser Testing
  - Expected Tokens
- Function Testing
  - Expected Output



Lessons Learned

- Good language design requires balance
- Impossible to stick to original specifications
- Details behind creating a language
  - Operator Precedence
  - Variable Scoping
  - Data type Manipulation
- Importance of Testing Early and Testing Often