Formal Languages
Final Project

Hiroki Hayashi
May 3rd, 2016

Formal Languages
Hiroki Hayashi
05/03/16

Objective:
Create a text editor that will alter the text-style of different parts of a program.

About the program:
GUI used: Swing and JEditorPane
This program follows a language specified in a DFA (shown below), that I have created. When keywords in the language are typed in the editor, those words should light up in different colors. On the console (in the IDE), it should tell you what key you pressed, and which state that key brought you to.

Demo:
The following will demonstrate how to use the text editor step-by-step.

1. To open the document, we must use window builder. I used Eclipse for this project, so to open it up on Eclipse, click File -> Import... -> General -> Existing Projects into Workspace -> Select root directory: (select the files in local machine) -> Finish.

   Once the project is imported to the workspace, all we need to do is to go to Package Explorer on the right column, right-click on the .java file and Open with... -> Window Builder. Run the ProjectFrame2.java file to start using the text editor.

2. The following window should pop up:

Formal Languages
Hiroki Hayashi
05/03/16
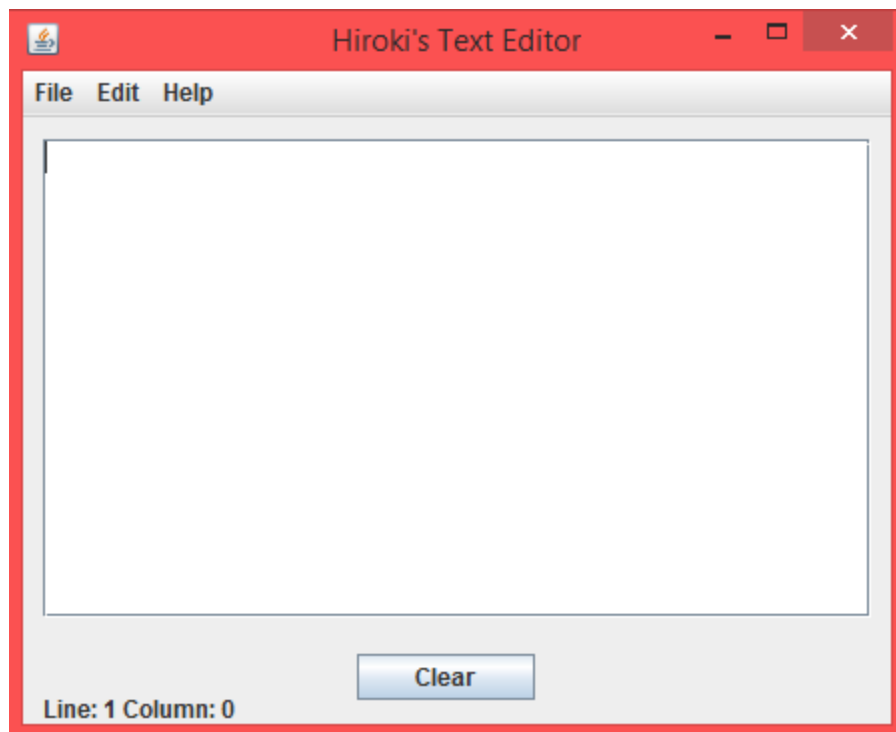
3. This is our text editor! We have a menu bar at the top, with File, Edit, and Help. The middle text area is where we type our functions (that light up in different colors), and the bottom contains a clear button and a status bar that shows you the line and column your cursor is currently at.

"File" menu contains:
- New – creates a blank slate
- Open File… - loads a file into the text editor
- Save – Saves the file
- Save – Saves the file into a file directory that the user specifies.
- Exit – closes the window

"Edit" menu contains:
- Change background – This changes the background color to white or black.

"Help" menu contains:
- Help – This should prompt the user on what help the user needs (not implemented due to time constraints).

4. Let's start typing something into the text editor. Our language is very small for now, so the keywords that we can put are limited.

Keywords:
var [char]
print ([char])
print ([int])
print (" [string] ")
char = [char]
char = [int]
char = " [string] "
char = [char] + [int]
- [CharList]-

Formal Languages
Hiroki Hayashi
05/03/16

5. Typing out each keyword gives us different colors. Unfortunately, because this is a JTextArea, we cannot have different colors for each string. However, we know what we're typing at the moment, and that is something.
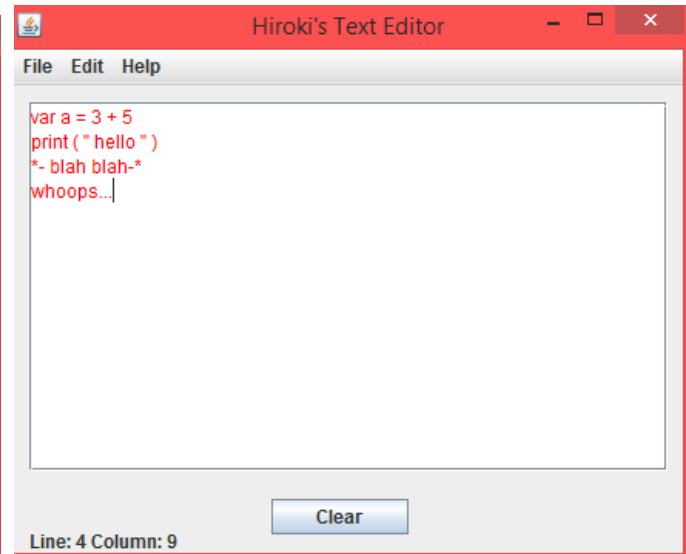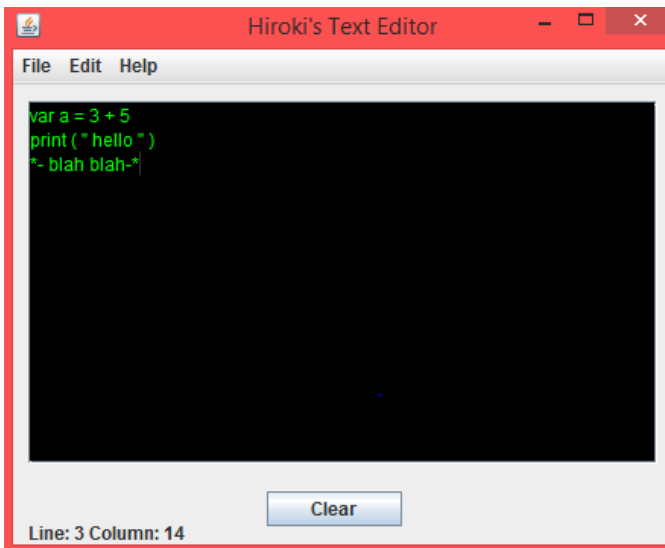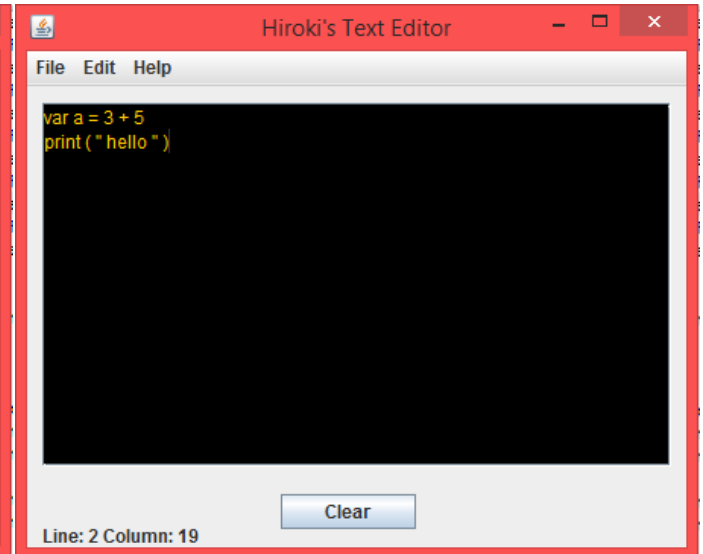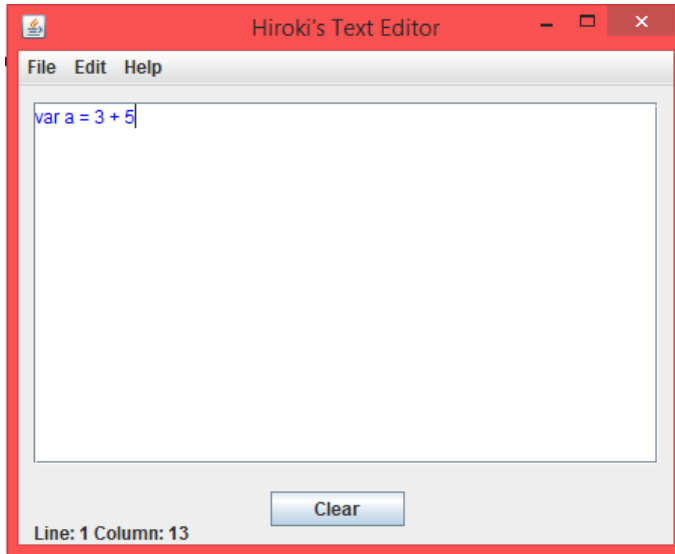
6. On the console, you might have realized something popping up. This is showing us what key we pressed, and what state we are in right now. Since this is checking every time we type, the running time becomes slow and it might not catch up to your typing speed.

```
Key Pressed: p
Current State: q1
Key Pressed: r
Current State: q3
Key Pressed: i
Current State: q4
Key Pressed: n
Current State: q4
Key Pressed: t
Current State: q5
Key Pressed: Space
Current State: q6
```

7. Restrictions:
   DFA :
   - cannot add variables in variable declarations i.e. var x = 3 +2
   - There is no way to get caret position on JTextPane, so this is in JTextArea.
     - Restrictions are currently just the color, where everything in the text editor must be the same color.
   - The comment does not accept a space after the character list.
     - e.g. Accepted: *- hello-*
       Unaccepted: *- hello -*