JFlap2Ti*k*Z

Andrew Mertz and William Slough

Introduction

JFlap is an excellent piece of software for experimenting with finite state machines, Turing machines and exploring many aspects of the theory of computation. JFlap2Ti*k*Z is a groovy script that converts a JFlap `jff` file representing a finite automaton, push down automaton, or Turing machine to LaTeX file depicting the automaton graphically using Ti*k*Z.

Requirements

To use JFlap2Ti*k*Z you will need Java installed. Additionally you may find it useful to have groovy installed as well. You will need to download either `JFlap2TikZ.jar` (if you only have Java installed) or `JFlap2TikZ.groovy` (if you also have groovy installed).

Usage

`JFlap2TikZ` is invoked from the command line, using Java

```
java -jar JFlap2TikZ.jar example.jff
```

or using groovy

```
groovy JFlap2TikZ.groovy example.jff
```

Note that in either case the output will be written to the console. Redirect the output or copy and paste as needed. Here is the full usage information.

```
usage: JFlap2TikZ [-g] [-h] [-r] [-s scale] [-z size] filename
 -g,--grid             Round positions so that they are on a grid
 -h,--help             Show usage information and quit
 -r,--rotate           rotate labels along edges
 -s,--scale <arg>      1 pixel in JFlap = scale points in LaTeX (default
                       is 1.0)
 -z,--gridsize <arg>   Set the spacing of the grid (default is 20.0)
```

Examples

The following figures show machines from the book JFLAP: An Interactive Formal Languages and Automata Package being converted to Ti*k*Z using JFlap2Ti*k*Z. Note that JFlap2Ti*k*Z currently ignores JFlap state annotations. Furthermore, not all JFlap files will be converted perfectly. You may have to adjust the scale and/or gridsize or edit the resulting Ti*k*Z code to achieve the effect you want. However, JFlap2Ti*k*Z should provide a good starting point.

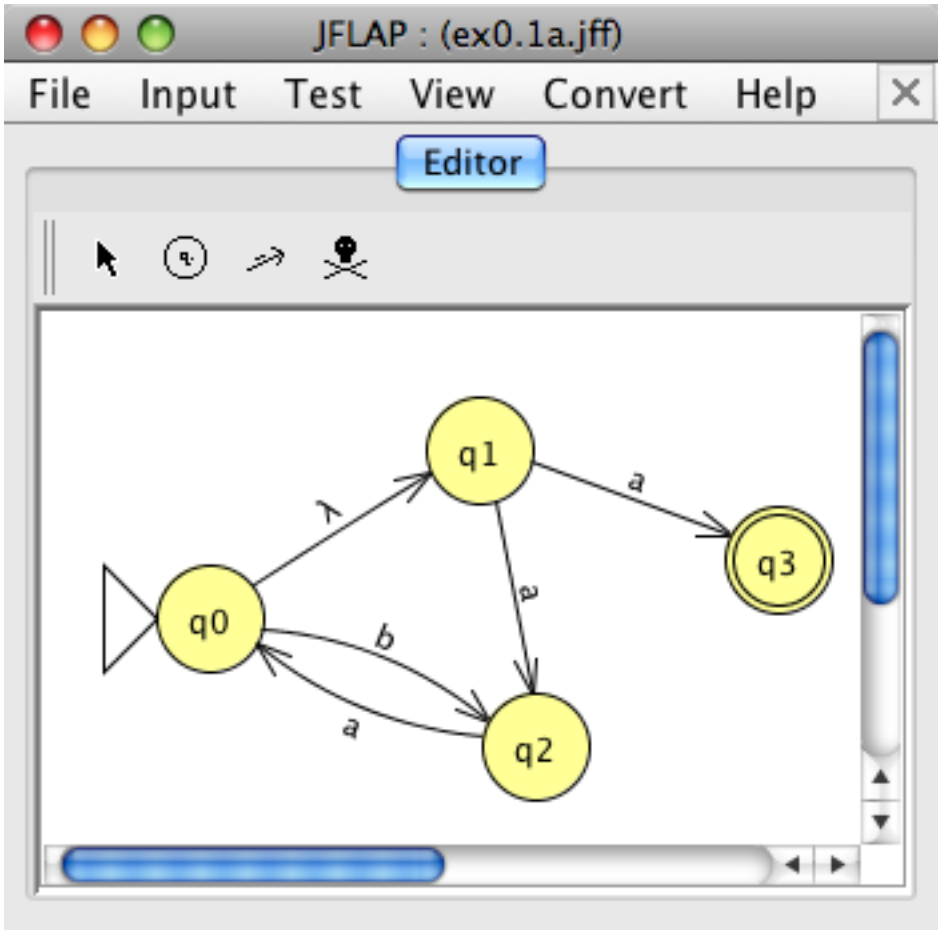

Figure 1: `ex0.1a.jff`

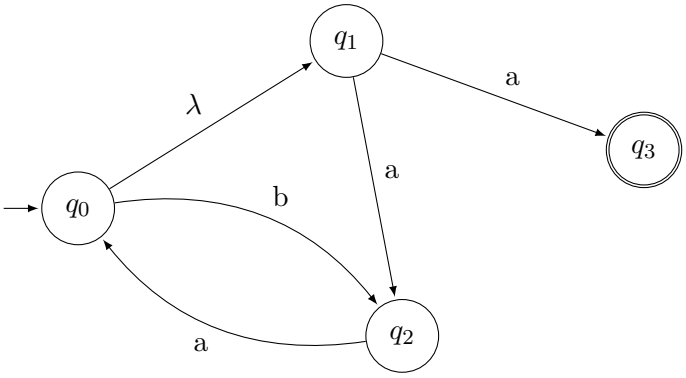

Figure 2: `ex0.1a.jff` converted to Ti*k*Z using default values

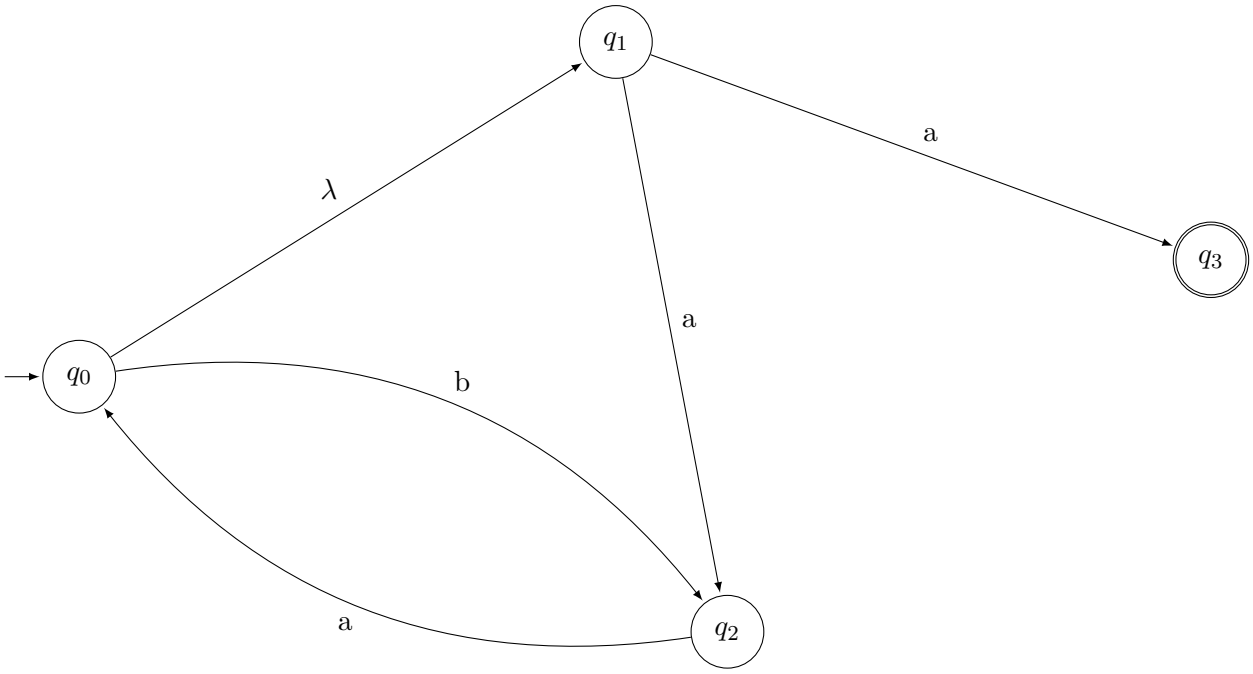

Figure 3: `ex0.1a.jff` converted to Ti*k*Z using a scale of 2

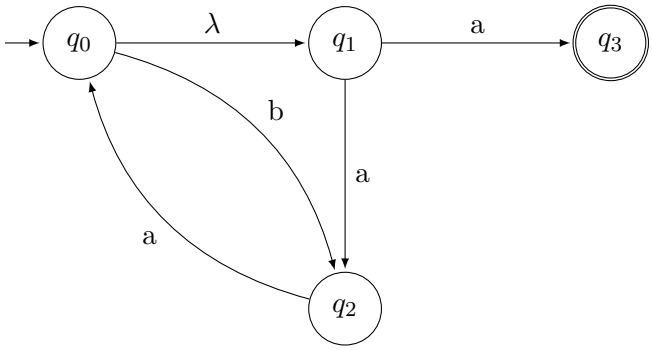

Figure 4: `ex0.1a.jff` converted to Ti*k*Z using a gridsize of 100

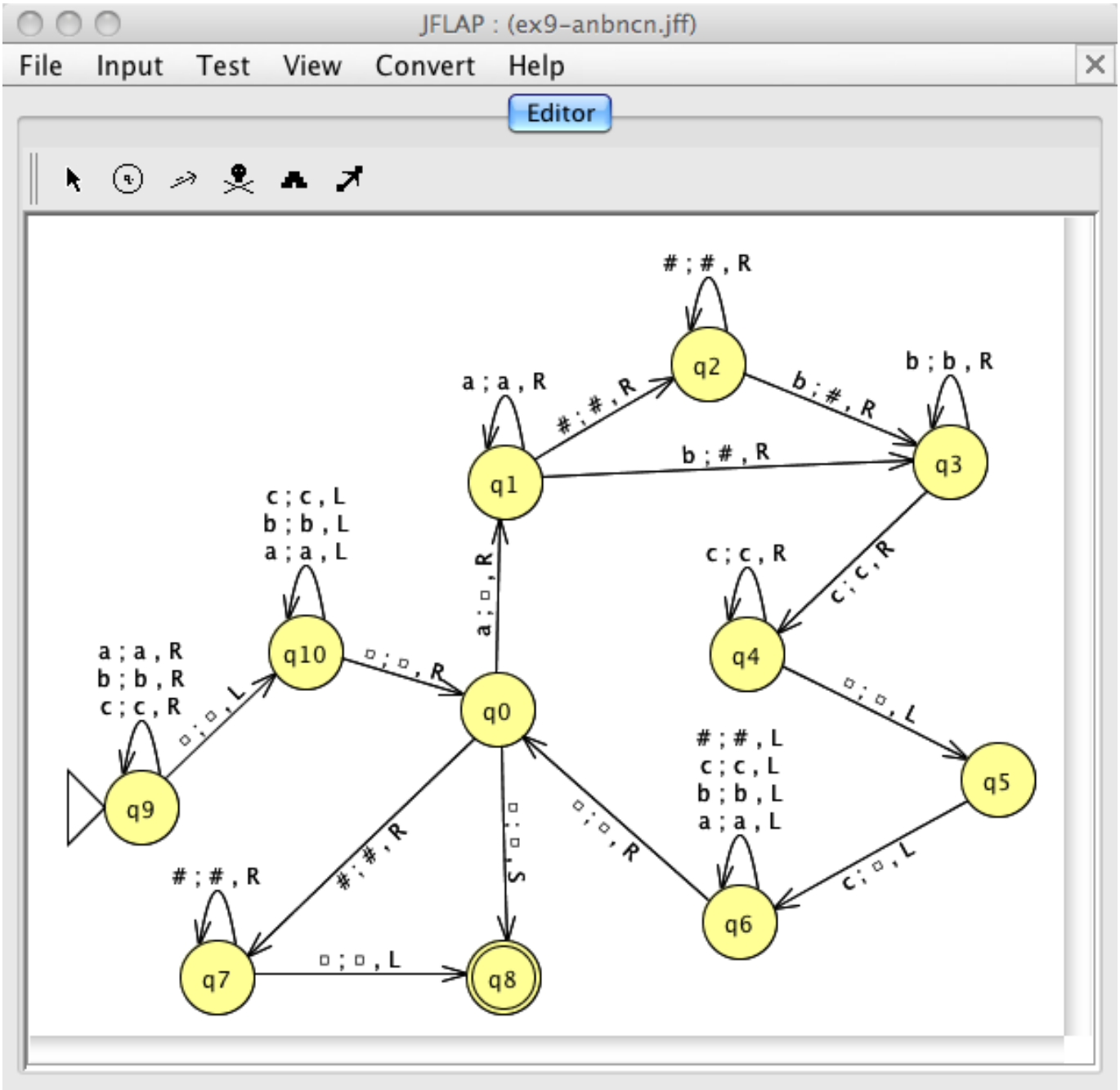

Figure 5: `ex9-anbncn.jff`

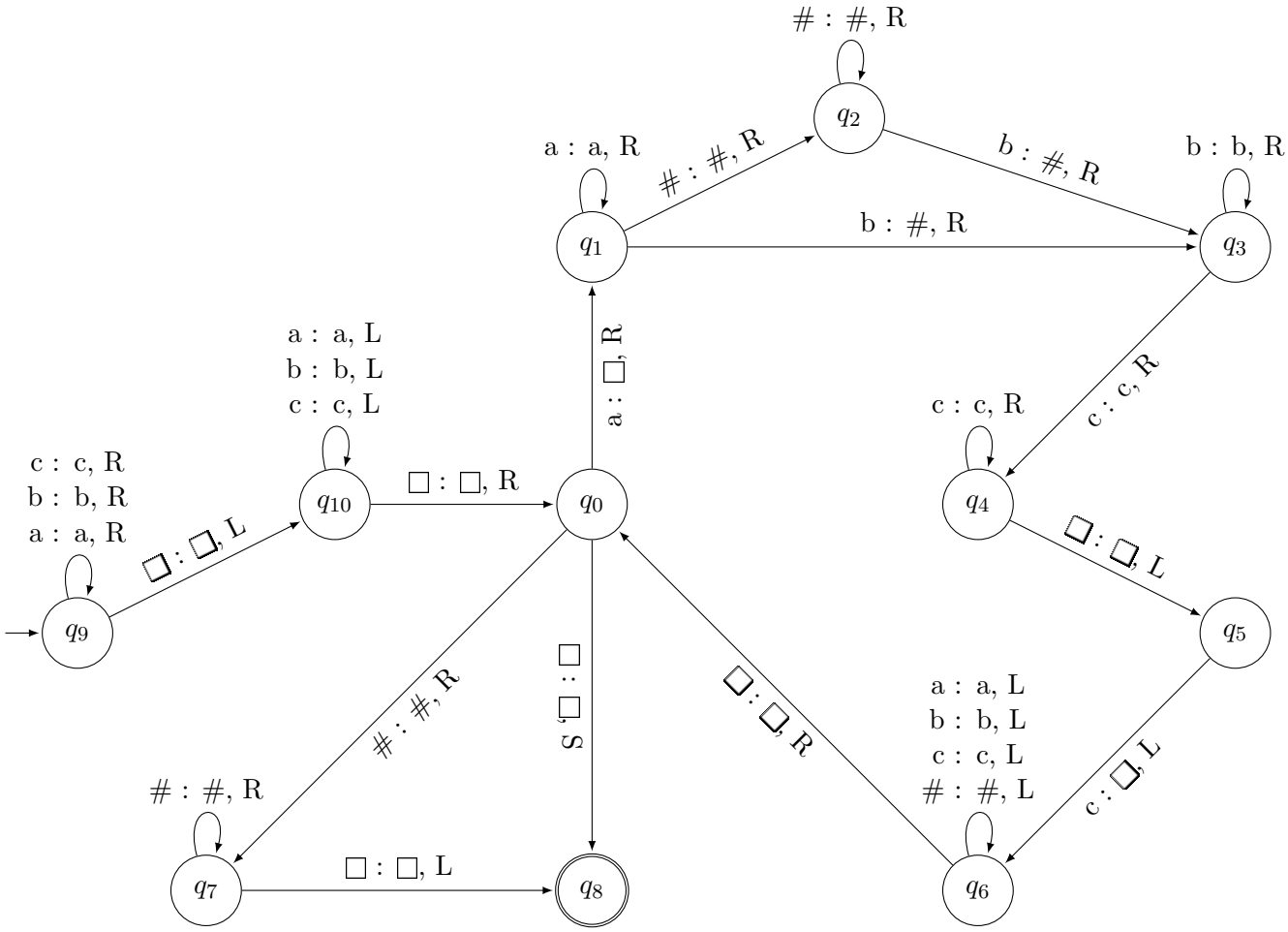

Figure 6: `ex9-anbncn.jff` converted to Ti*k*Z using a gridsize of 50 and label rotations on

License

JFlap2Ti*k*Z and its documentation are licensed under an MIT style license