Converting a Regular Grammar to DFA

Jay Bagga

1 Regular Grammar to DFA

In this exercise we use JFLAP to convert a regular grammar to a DFA. The regular (right linear) grammar that we use as our example is shown in Figure 1. Input this grammar into JFLAP.

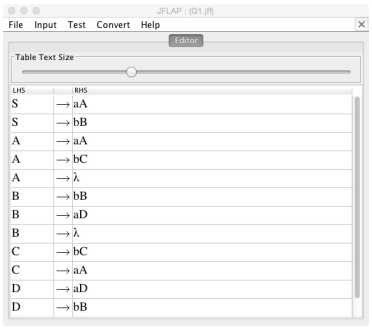

Figure 1: Input Regular Grammar

The algorithm to convert a regular grammar to an FA is straightforward. We create a state for every variable and one extra final state. The state corresponding to the start variable is the initial state. For every production $X \to xY$ we draw a transition from state X to state Y and label it x. For every production $X \to w$, we create a transition from state X to the final state and label it w.

Select Convert to FA. JFLAP creates the states for each variable and one final state as shown in Figure 2.

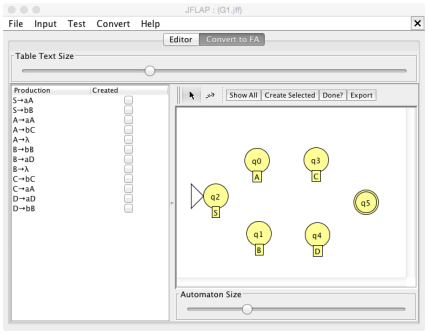

Figure 2: Variables to States

Add transitions corresponding to the productions $S \to aA$ and $S \to bB$ as shown in Figure

3. Repeat this for the remaining productions. The FA so obtained is shown in Figure 4.

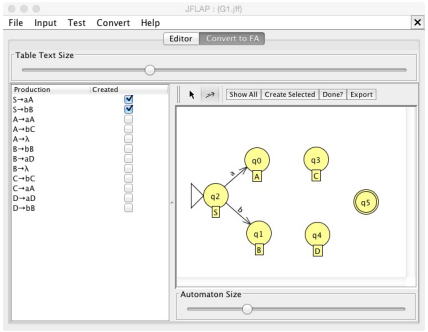

Figure 3: Productions to Transitions

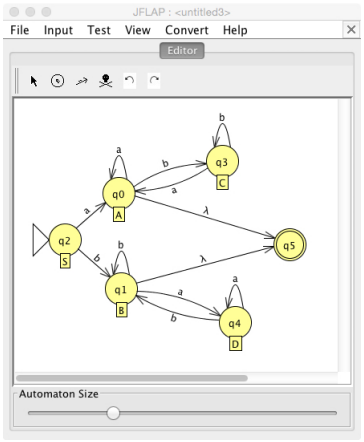

Figure 4: Completed FA

Finally we can use JFLAP to convert this FA to a DFA. The result is shown in Figure 5. What language does this DFA accept? Is this the same language that is generated by the regular grammar you started with?

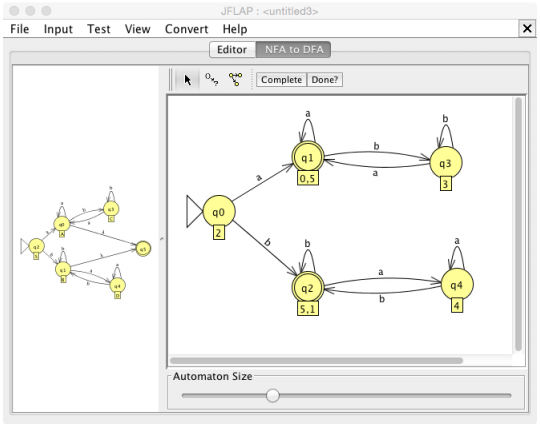

Figure 5: Completed DFA

2 References

1. Introduction to the Theory of Computation (Third Edition), Michael Sipser. Cengage Learning. 2013.

2. JFLAP - An Interactive Formal Languages and Automata Package, Susan H. Rodger and Thomas W Finley. Jones and Bartlett Publishers. 2006