

2020

Методические указания к выполнению лабораторных работ по курсу «Дискретная математика» в среде JFLAP для специальностей

Приведены сведения, необходимые для выполнения лабораторных работ
по дисциплине «Дискретная математика»

26.03.2020

А. В. Саяпин
СибГАУ



Введение

Формальные грамматики

Теория формальных грамматик находит широкое применение в различных областях компьютерных наук. В качестве примера можно назвать использование регулярных выражений, как в области обработки текста, так и в области компьютерной безопасности.

Рассмотрим примеры работы с формальными грамматиками при помощи приложения JFLAP.

JFLAP представляет собой программный комплекс, написанный на языке Java, и реализующий механизмы для работы с формальными грамматиками, конечными автоматами, регулярными выражениями и машинами Тьюринга. Комплекс распространяется с открытыми исходными кодами под лицензией GNU GPL.

Запуск JFLAP

Система JFLAP может быть запущена на любой платформе, поддерживающей запуск Java-приложений. Для запуска необходима среда JAVA RTE версии не ниже 1.5.

Для запуска в среде Windows достаточно выполнить двойной щелчок по файлу системы (jflap.jar). Для запуска в среде Unix/Linux необходимо выполнить команду

```
java -jar jflap.jar
```

в командной строке. В Unix/Linux команды чувствительны к регистру символов.

При запуске возникает основное окно программы (рис. 1).



Рис. 1. Основное окно программы.

Общие действия с грамматикой

Часто возникает необходимость сохранить введенную грамматику для дальнейшего использования, либо использовать ранее созданную грамматику. Кроме того, дерево синтаксического разбора может потребоваться для дальнейшей работы как иллюстрация.

Для выполнения таких задач можно использовать пункт меню *File*. Он включает в себя такие подпункты, как

New... – создает новую грамматику;

Open... – загружает ранее сохраненную грамматику;

Save – сохраняет текущую грамматику;

Save As... – сохраняет текущую грамматику в указанный файл;

Save Image As... – сохраняет текущую грамматику и ее разбор в указанный файл в виде изображения.

Грамматики сохраняются в формате XML, что позволяет при необходимости редактировать их даже в отсутствие системы JFLAP.

Редактирование грамматики

JFLAP предназначен для работы с контекстно-свободными и автоматными грамматиками.

Грамматика в JFLAP, как и в теории формальных грамматик, рассматривается как совокупность четырех объектов:

$G = \{ V_m, V_A, \langle I \rangle \in V_A, R \}$, где

V_m – терминальный алфавит,

V_A – нетерминальный алфавит,

$\langle I \rangle$ – начальный символ грамматики (ϵ),

R – множество правил грамматики, называемых продукциями (Productions).

Для перехода в режим редактирования грамматики в основном окне программы необходимо выбрать пункт *Grammar*. В результате откроется окно редактирования грамматики (рис. 2).

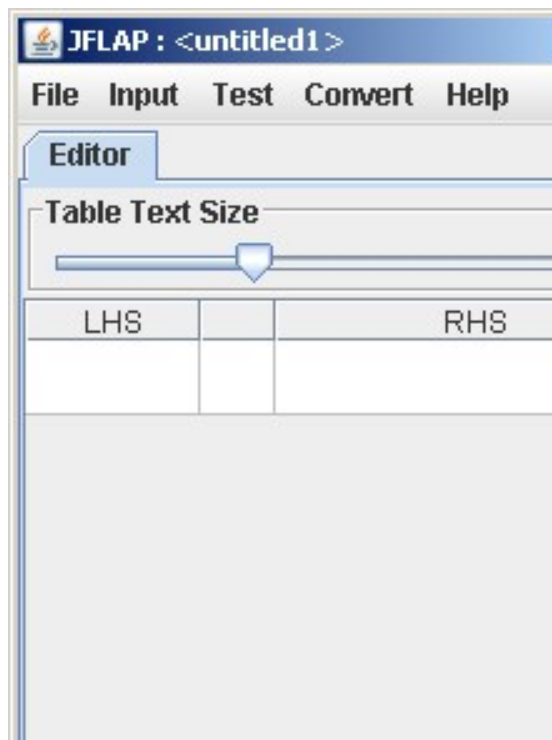


Рис. 2. Окно редактирования грамматики.

Окно содержит таблицу, левый столбец которой содержит левые части правил грамматики (LHS), а правый столбец таблицы – правые части правил грамматики (RHS). Регулятор *Table Text Size* позволяет изменять размер шрифта в правилах. Следует учитывать, что левые части правил не могут содержать несколько символов, и не могут содержать нетерминальные символы (таковы особенности регулярных и контекстно-свободных грамматик).

При вводе грамматик необходимо учитывать следующее:

- JFLAP воспринимает все одиночные символы, набранные строчными буквами, как терминальные символы;
- JFLAP воспринимает все **одиночные** символы, набранные прописными буквами, как нетерминальные символы;
- JFLAP воспринимает как начальный символ любой символ, находящийся в левой части **первого** правила грамматики.

Для перемещения по таблице можно использовать клавишу TAB на клавиатуре.

Для ввода пустого правила достаточно ввести левую часть правила, а правую часть правила оставить пустой. JFLAP автоматически заменит пустую часть правила на символ ϵ .

Разбор строки в соответствии с правилами грамматики путем простого перебора

JFLAP реализует различные способы разбора слова языка, описанного формальной грамматикой. Один из самых простых в алгоритмическом смысле способов разбора – это брутфорс (Brute-Force), то есть разбор слова простым перебором. В этом случае JFLAP пытается последовательно заменять начальный символ грамматики последовательностями символов в соответствии с правилами грамматики.

Для разбора строки простым перебором, из режима редактирования грамматики необходимо вызвать пункт меню *Input/Brute Force Parser*.

В результате на экране появится окно разбора (рис. 3).

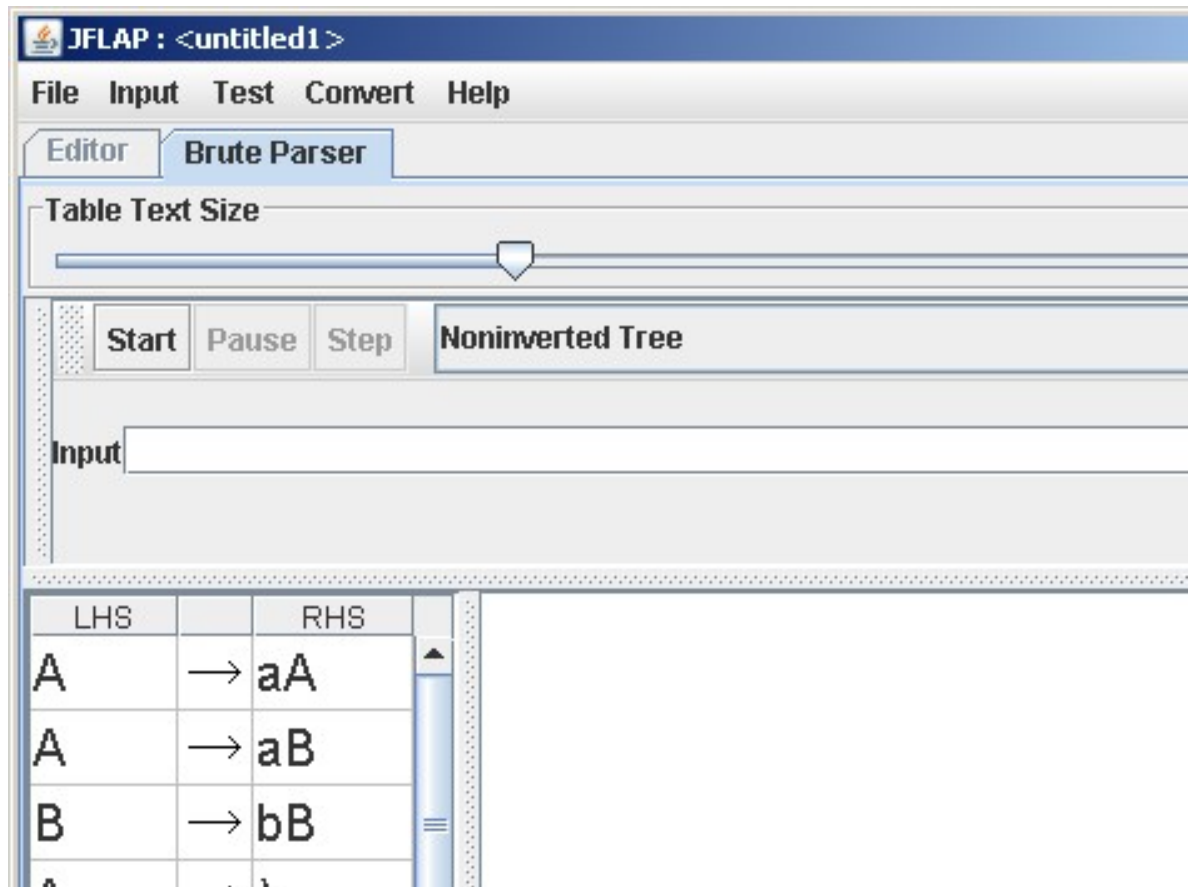


Рис. 3. Окно разбора строки путем перебора.

Для ввода анализируемой строки служит поле *Input*.

В левой части окна перечислены правила грамматики, а в правой демонстрируется последовательность вывода строки из начального символа грамматики. Вывод может быть показан либо в виде дерева синтаксического анализа (*Noninverted Tree*), либо в виде последовательности промежуточных строк, содержащих как терминальные, так и нетерминальные символы (*Derivation table*).

Для выполнения разбора служит кнопка *Start*. После ее нажатия JFLAP сообщит, принадлежит ли строка заданному языку (*string accepted*), или нет (*string rejected*), а так же количество узлов в дереве разбора.

Для последовательного построения дерева анализа или таблицы вывода служит кнопка *Step*. В строке статуса JFLAP показывает, какое правило было применено.

После выполнения разбора кнопка *Step* становится недоступна.

Кроме того, существует возможность разбора нескольких строк за один раз. Для этого служит пункт меню *Input/Multiple Brute Force Parse* (рис. 4).

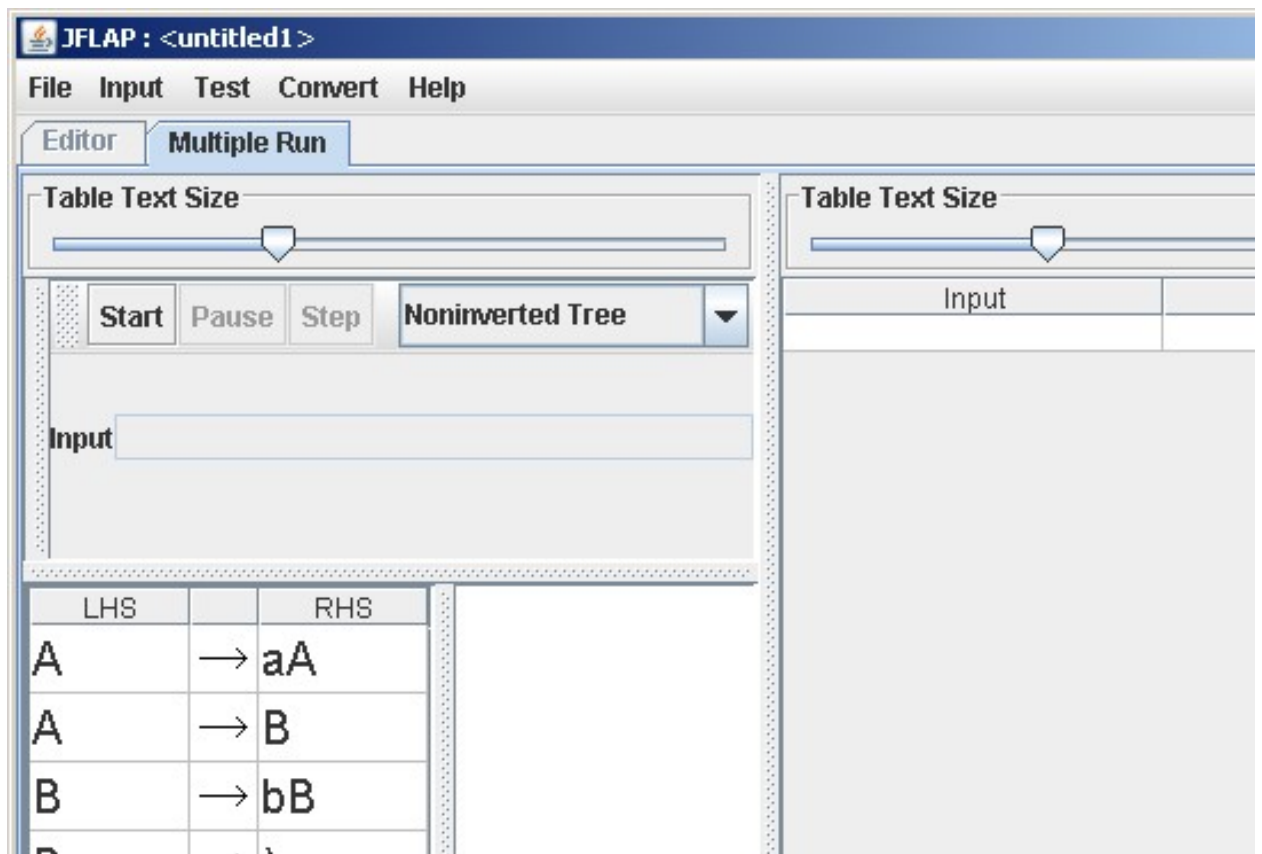


Рис. 4. Разбор нескольких строк.

В этом случае окно программы делится на две части. Левая часть соответствует разбору строки простым перебором, а в правой части задаются строки для разбора (*Input*) и результаты разбора (*Result*). Левая часть окна в этом режиме недоступна для редактирования.

В правой части окна так же располагаются несколько кнопок, выполняющих специальные действия. *Load Inputs* позволяет загрузить список строк для разбора из указанного файла, *Run Inputs* запускает разбор всех строк, *Clear* очищает список строк для разбора, *Enter Lambda* позволяет ввести пустую строку. Последнее необходимо для проверки того, принадлежит ли пустая строка языку, определенному данной грамматикой.

Определение типа грамматики

В JFLAP существует возможность определить тип введенной грамматики. Для этого необходимо в окне редактирования грамматики выбрать пункт меню *Test/Test for Grammar Type*. JFLAP определит тип грамматики, введенной пользователем.

Преобразования грамматик

Преобразования формальных грамматик достаточно часто встречаются в практической деятельности. Такие преобразования могут, например, потребоваться после разработки грамматики, описывающей какой-либо язык. Это вызвано тем, что при создании грамматики ее разработчик описывает ее в виде, удобном именно ему (например, вводя дополнительные нетерминалы, описывающие какие-либо понятия предметной области). Грамматика, используемая для обработки реального языка, часто должна быть представлена в некой стандартной форме, например, нормальной форме Хомского, или нормальной форме Грейбах.

Преобразования грамматик мы будем рассматривать в приложении к контекстно-свободным грамматикам на примере преобразования в нормальную форму Хомского.

Удаление пустых правил.

К пустым, или аннулирующим правилам относят правила контекстно-свободной грамматики, имеющие вид $A \rightarrow \epsilon$, в языке, не содержащем пустую строку.

Удаление осуществляется путем выявления нетерминальных символов, из которых может быть выведена пустая строка и замене правил, ведущих к пустой строке, на эквивалентные.

Для преобразования грамматики необходимо ввести ее в систему JFLAP каким-либо образом (см. п. Редактирование грамматики), и выбрать пункт меню *Convert/Transform Grammar*. В результате на экране появится окно удаления пустых правил (рис. 5).

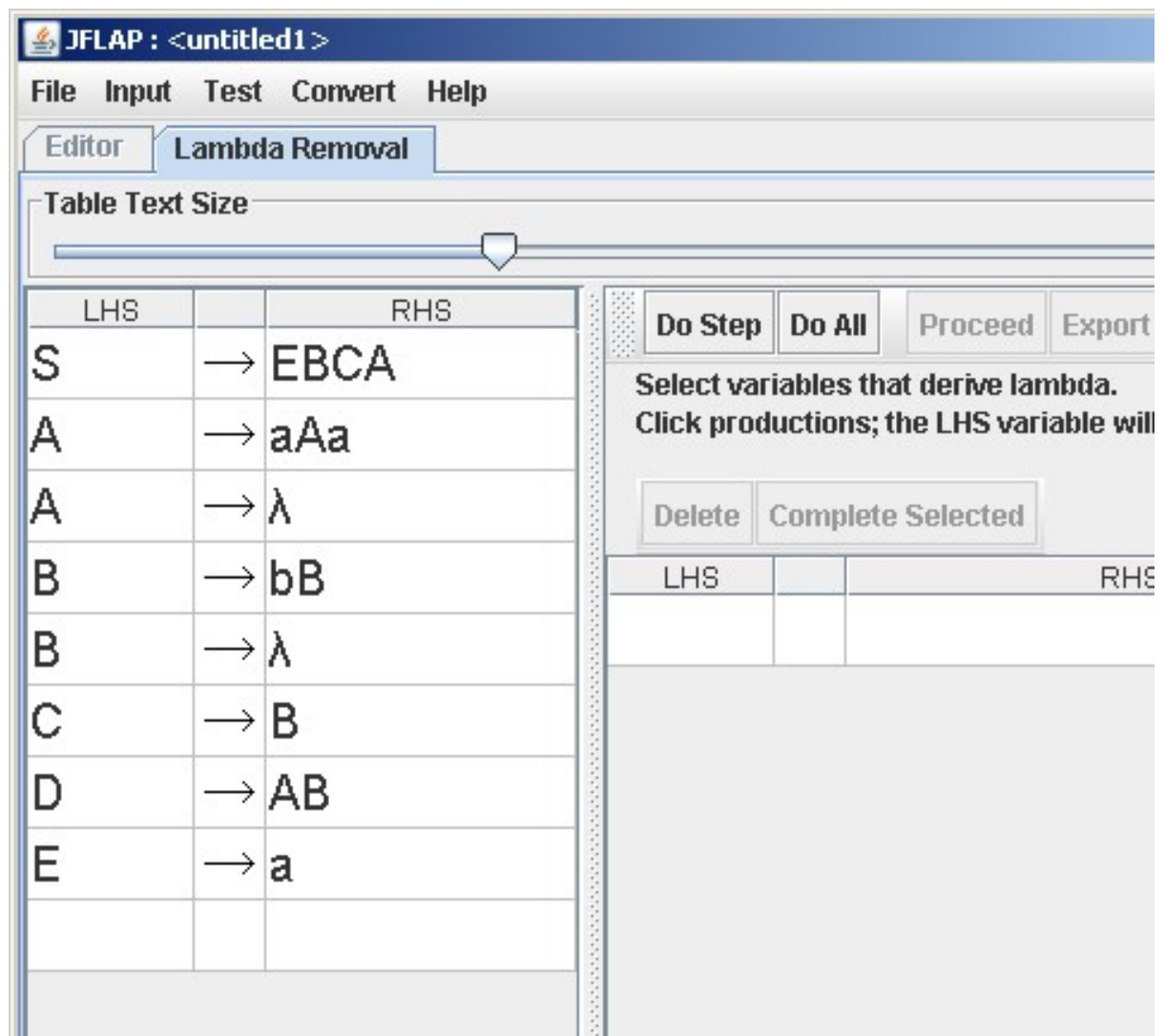


Рис. 5. Окно удаления пустых правил контекстно-свободной грамматики.

В левой части окна представлена исходная грамматика, а в правой части – преобразованная. Удаление правил, приводящих к пустой строке, может быть выполнено либо в автоматическом, либо в ручном режиме.

Удаление пустых правил в автоматическом режиме

Для преобразования грамматики в автоматическом режиме необходимо использовать кнопку *Do Step*. В результате будут удалены правила, приводящие к пустой строке, и будут добавлены новые правила, для того, чтобы новая грамматика была эквивалентна исходной (рис. 6).

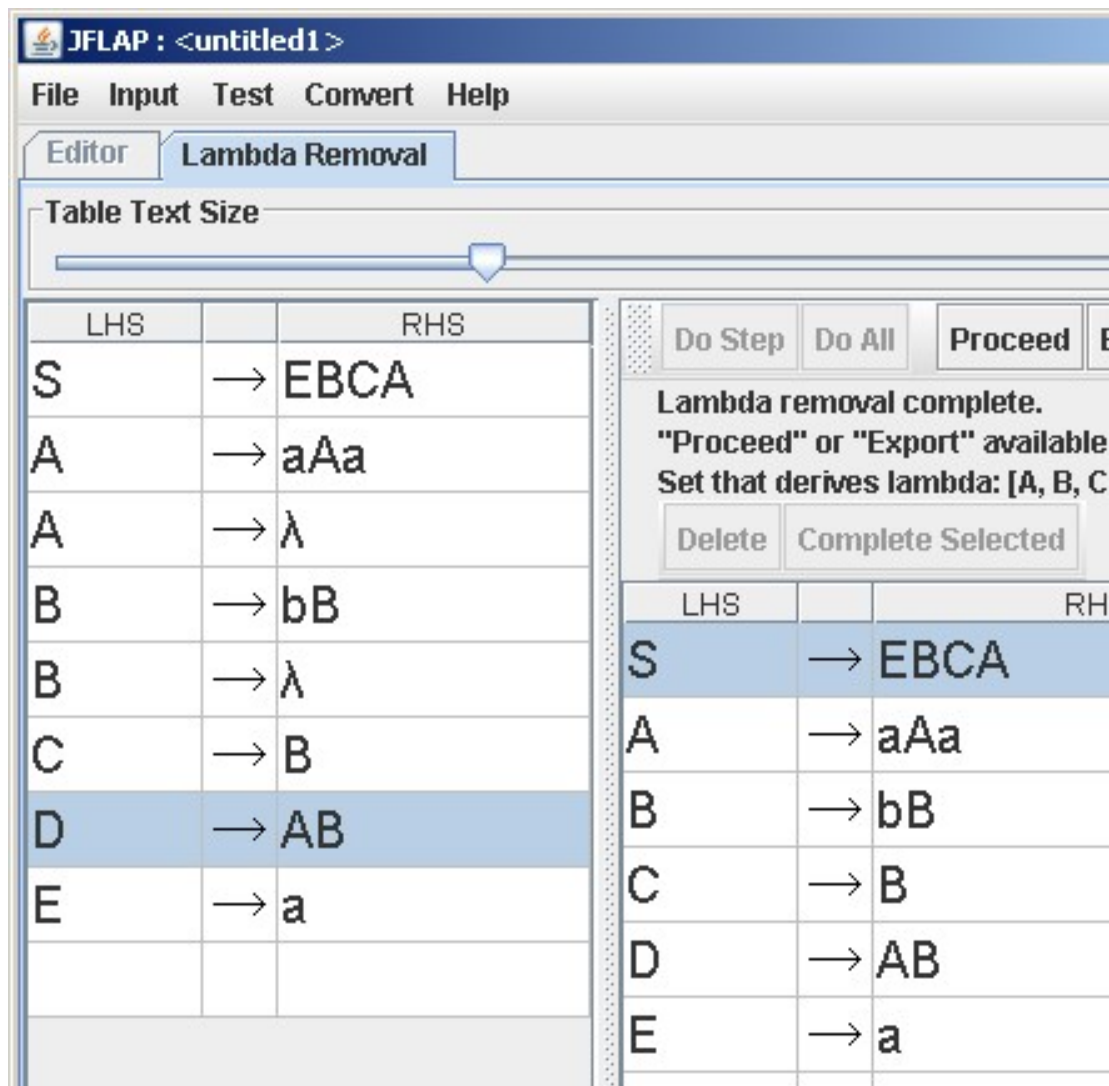


Рис. 6. Удалены правила, приводящие к пустой строке.

Удаление пустых правил в ручном режиме

Для преобразования грамматики в ручном режиме необходимо выполнить несколько шагов.

Прежде всего, необходимо выделить нетерминальные символы, из которых могут быть выведены пустые строки. Выделение таких нетерминальных символов осуществляется выбором соответствующих правил из левой части окна. Например, пустая строка может быть выведена из нетерминального символа A. Выберем правило $A \rightarrow \epsilon$ из левой части окна (рис. 7).

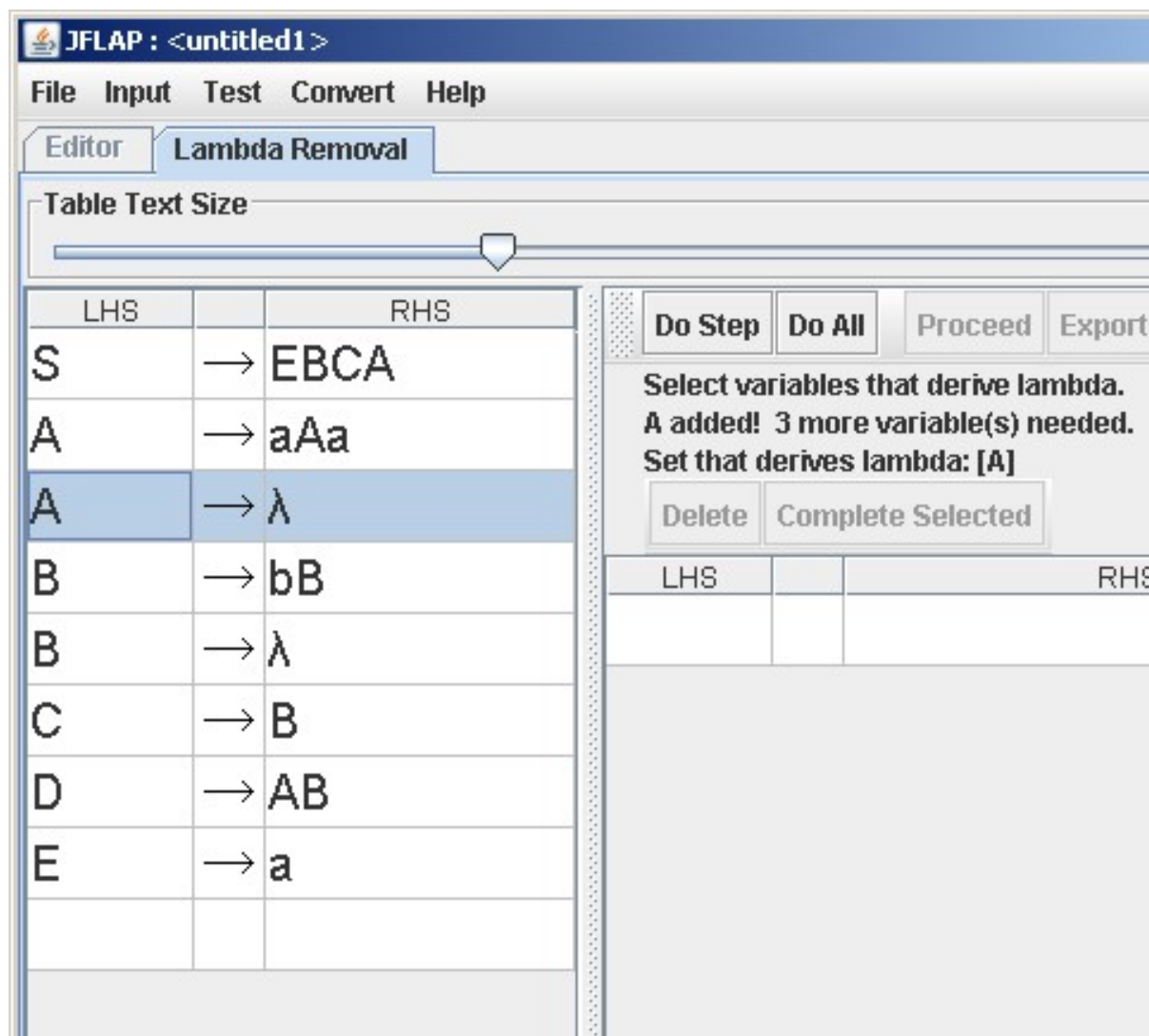


Рис. 7. Добавление переменной, приводящей к пустой строке.

В правой части окна после выбора правила возникает подсказка, указывающая, что имеется еще 3 переменные, которые необходимо выбрать (*3 more variable(s) needed*).

Следующим шагом, очевидно, можно выбрать переменную B (правило $B \rightarrow \epsilon$). Так же пустая строка может быть выведена из нетерминального символа C (исходя из правила $C \rightarrow \epsilon$). Последним искомым нетерминальным символом является D, из которого могут быть выведены символы A и B, из которых выводятся пустые строки.

После выбора последнего из нетерминальных символов, из которого может быть выведена пустая строка, окно меняется (рис. 7).

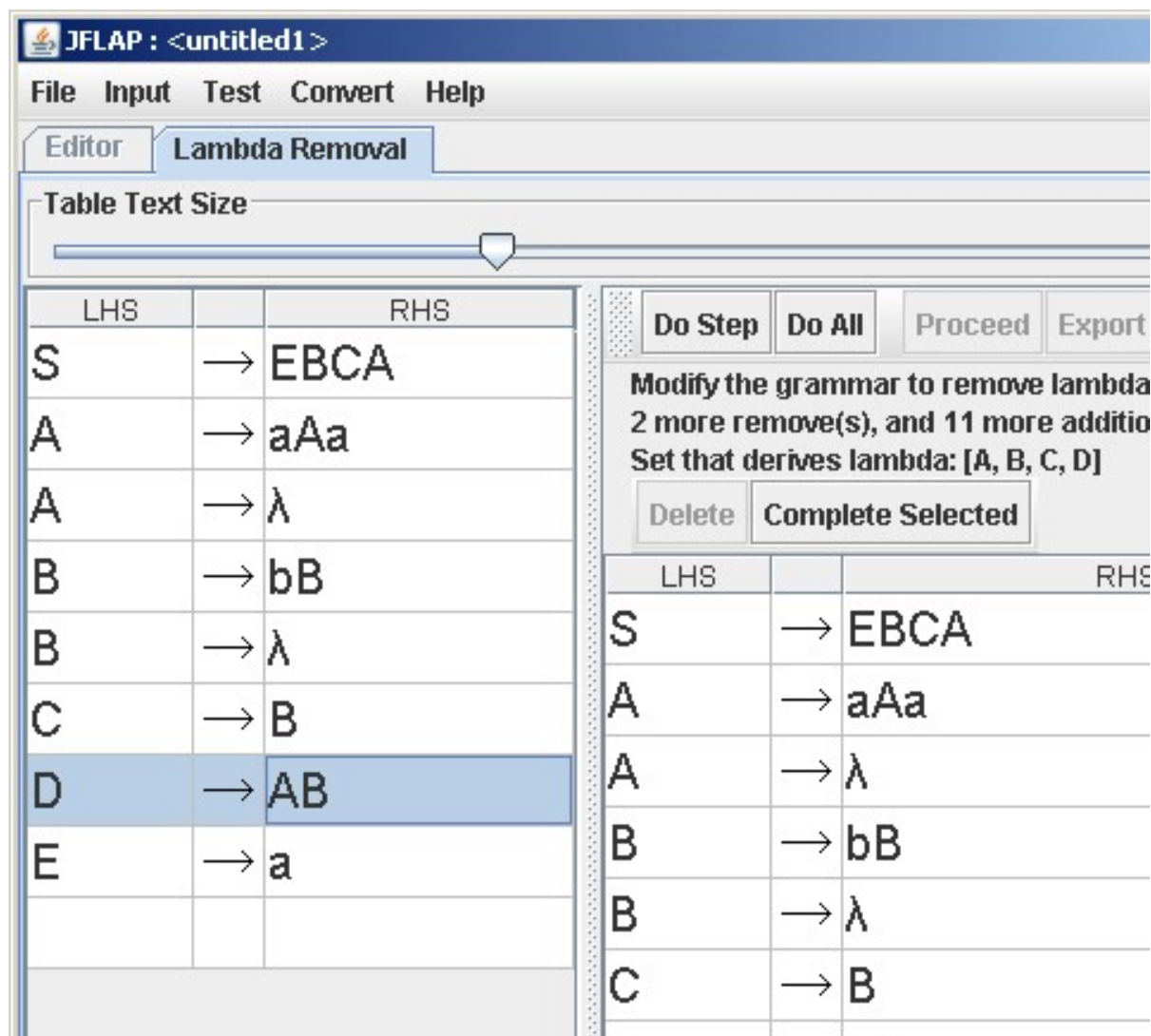


Рис. 7. Выбор нетерминальных символов закончен.

JFLAP указывает, что множество искомых нетерминалов содержит 4 символа, A, B, C, D: *Set that derives lambda [A, B, C, D]*.

Для получения грамматики, эквивалентной исходной, необходимо введение дополнительных правил, компенсирующих отсутствие пустых правил. JFLAP подсказывает минимально необходимое количество новых правил: *and 11 more addition(s) needed*.

Правила могут быть добавлены вручную в правую часть окна грамматики, либо добавлены системой JFLAP автоматически, путем нажатия на кнопку *Complete Selected*, либо *Do Step*, либо *Do All*.

Удаление правил с единственным нетерминальным символом в правой части правил.

После удаления пустых правил можно переходить к удалению правил с единственным нетерминальным символом в правой части. Для этого необходимо нажать кнопку *Proceed* (она становится доступной после удаления пустых правил, рис. 8).

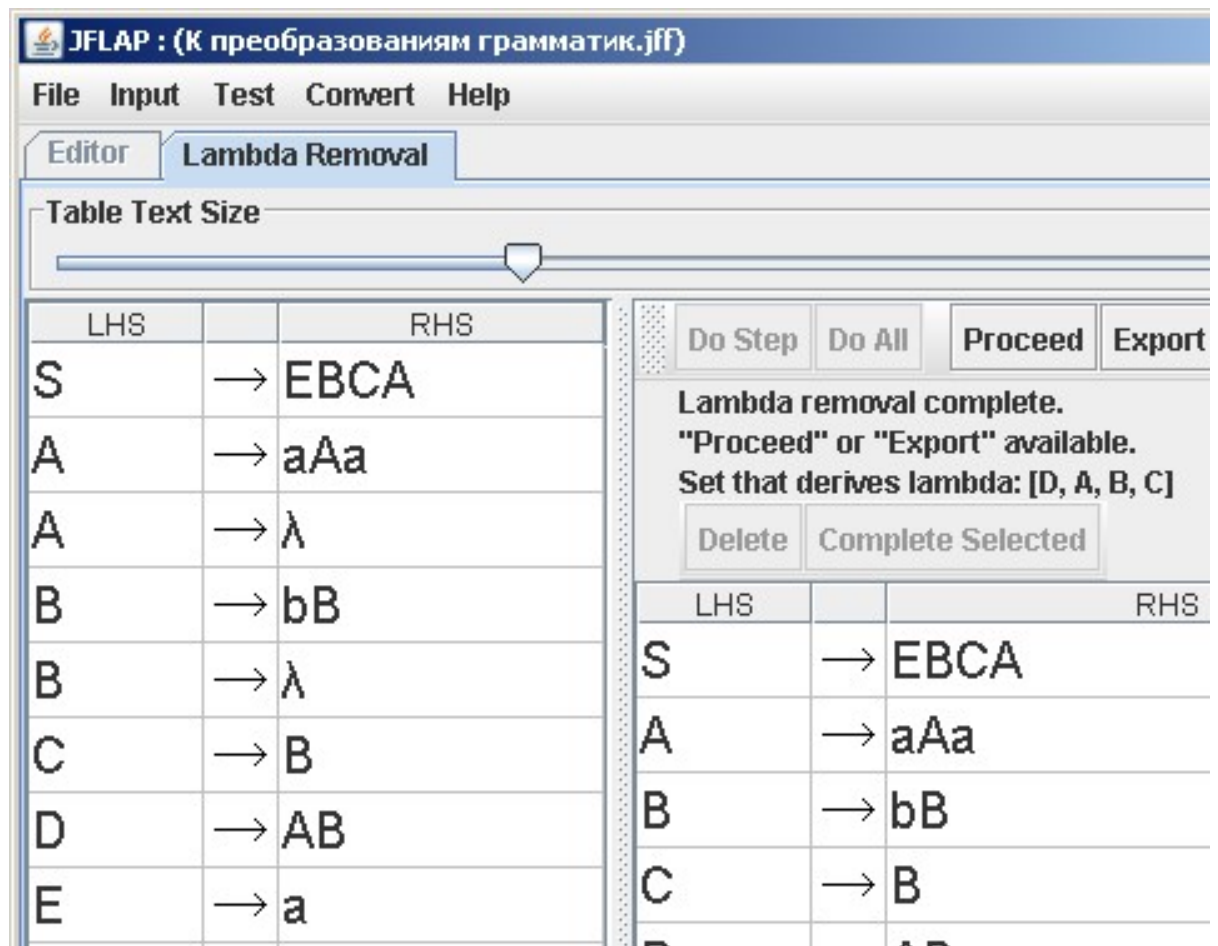


Рис. 8. Удаление правил с единственным нетерминальным символом в правой части.

В результате мы переходим в режим удаления правил (рис. 9).

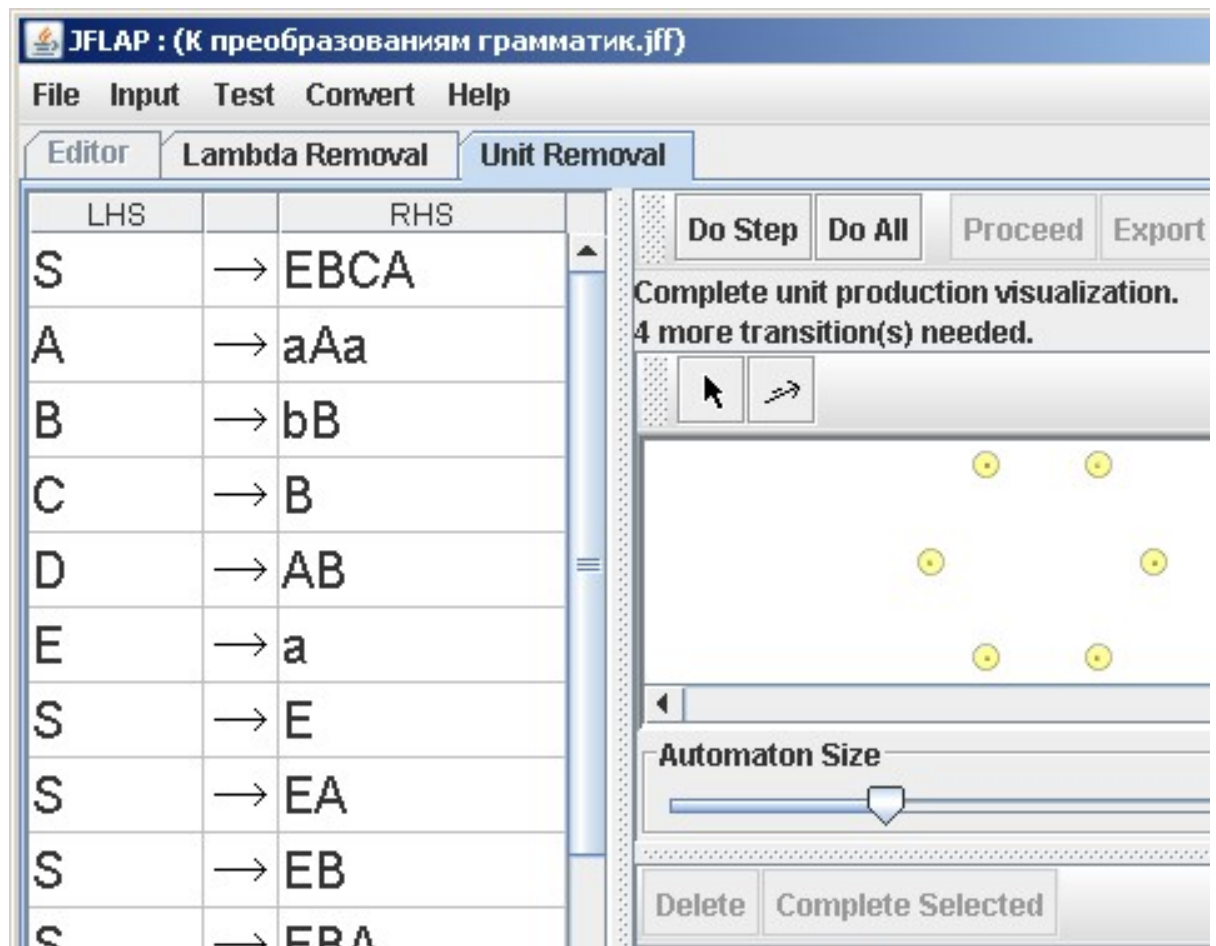


Рис. 9. Режим удаления правил с единственным нетерминальным символом в правой части.

Кнопка *Do All* автоматически выполнит все необходимые действия. Кнопка *Do Step* выполнит весь процесс по шагам.

Автоматы

JFLAP имеет средства для работы с детерминированными и недетерминированными конечными автоматами, а так же детерминированными и недетерминированными автоматами с магазинной памятью. Рассмотрим работу с ними подробнее.

Конечные автоматы

Для работы с детерминированными и недетерминированными автоматами в основном меню JFLAP необходимо выбрать пункт *Finite Automaton* (рис. 10).



Рис. 10. Выбор режима работы с конечными автоматами.

В результате откроется окно редактирования автомата.

Редактирование конечного автомата

Конечный автомат может быть создан или отредактирован в окне редактора (рис. 11). Для

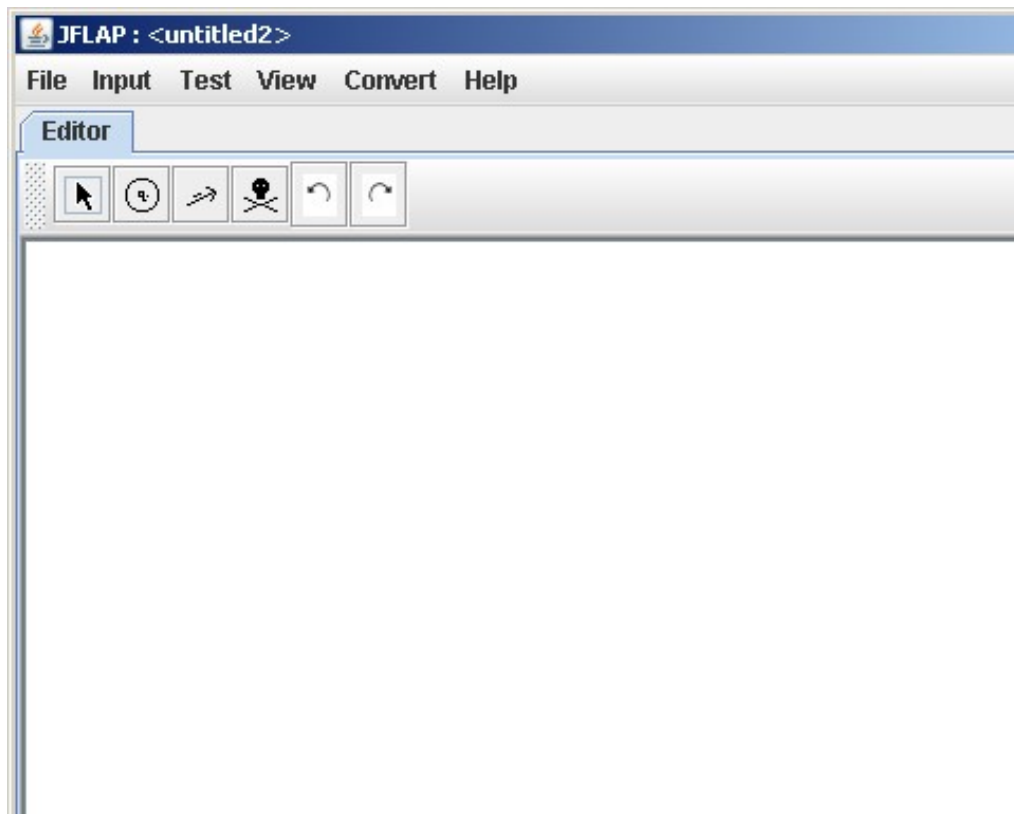


Рис. 11. Окно редактирования конечного автомата.

создания состояний автомата требуется выбрать кнопку  на панели инструментов. Выбранный инструмент затемняется. При выбранном инструменте создания состояний каждый щелчок на свободном поле приведет к созданию нового состояния автомата.

Для создания переходов между состояниями используется кнопка . Для создания перехода из состояния q_i в состояние q_j по символу a необходимо нажать кнопку на состоянии q_i и протянуть курсор мыши на состояние q_j . Появится стрелка перехода с полем для указания символа перехода. Если необходимо создать переход по пустому символу, поле необходимо оставить пустым, щелкнув мышью на свободном поле.

Для удаления состояний или переходов используется кнопка .

Для перемещения состояний по изображению, переименования состояний используется кнопка . Нажатие на правую кнопку мыши (рис. 12) позволяет пометить состояние как начальное (*Initial*), заключительное (*Final*), изменить пометку состояния (*Change Label*), удалить все наименования состояний (*Clear All Labels*) и задать имя состояния (*Set Name*). Пометки состояний отображаются в нижней части кружка, изображающего состояние. Нажатие правой кнопкой мыши на свободном пространстве изображения позволяет автоматически расположить состояния автомата наиболее удобным образом (*Layout Graph*), добавить заметку (*Add Note*), включить режим автоматического масштабирования (*Auto Zoom*).

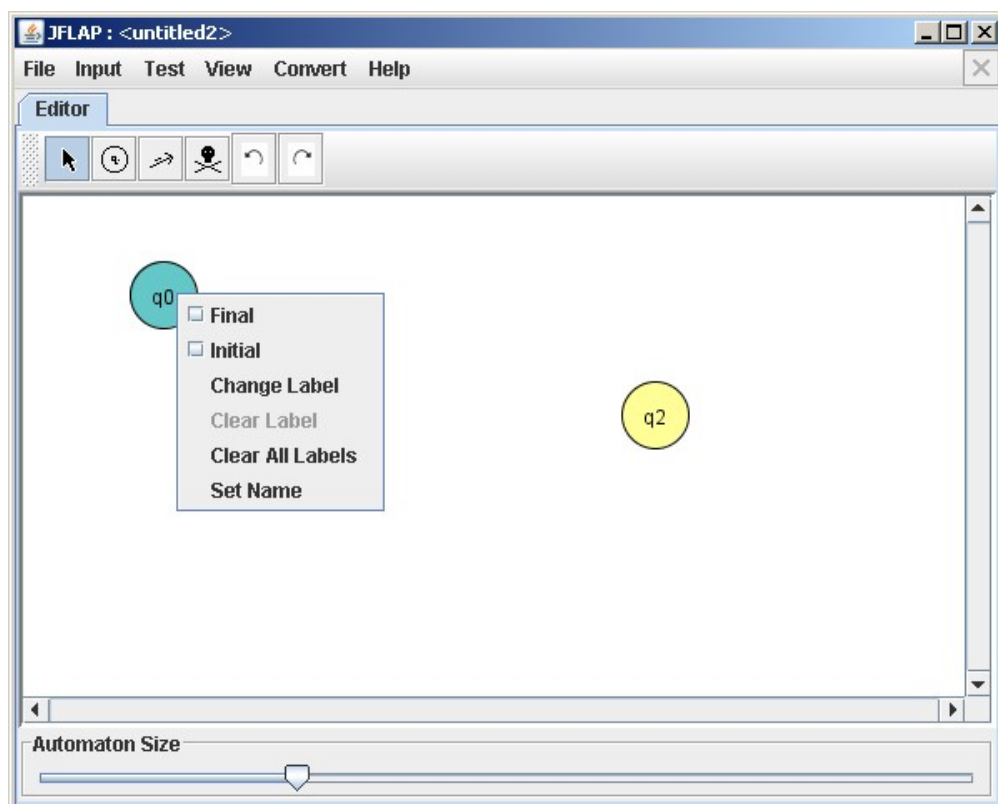


Рис. 12. Свойства состояния.

Разбор строки автоматом

Для разбора строки автоматом необходимо воспользоваться пунктом меню *Input/Fast Run....* На экране появится окно, показанное на рис. 13.

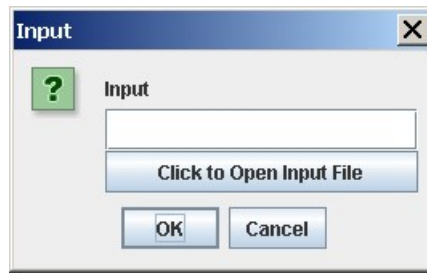


Рис. 13. Окно ввода строки

Строка может быть загружена из текстового файла с использованием кнопки *Click to Open Input File*. После ввода строки для распознавания нажмите кнопку *OK*. В случае, если строка не распознана автоматом, появится окно, показанное на рис. 14.



Рис. 14. Окно, показывающее, что строка не распознана автоматом.

В противном случае, то есть, если автомат распознал данную строку (находится в одном из допустимых заключительных состояний), будет показана последовательность конфигураций автомата (рис. 15).

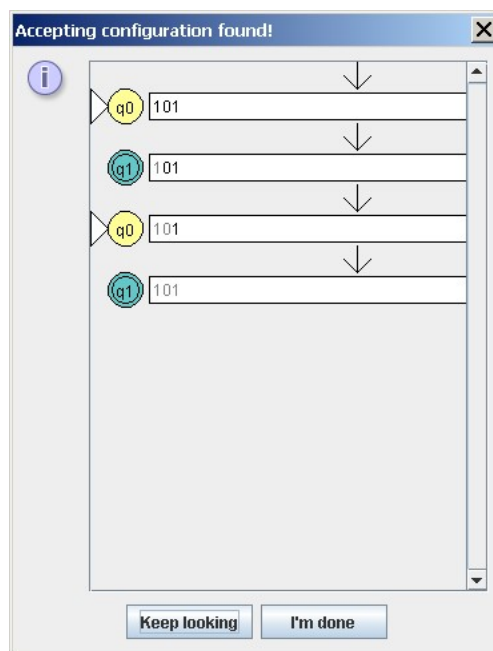


Рис. 15. Последовательность конфигураций автомата, распознавшего строку.

Уже прочитанные символы строки помечаются серым цветом. Нажатие на кнопку *I'm done* приводит к закрытию окна конфигураций. Нажатие на кнопку *Keep looking* приведет к выводу сообщения о том, что строка распознана.

Определение недетерминированности автомата и ϵ -переходов

Для определения того, является ли автомат недетерминированным, необходимо выбрать пункт меню *Test/Highlight Nondeterminism*. В результате будет открыта новая закладка, в которой будут подсвечены переходы в разные состояния по одинаковым символам.

Для определения того, имеет ли автомат переходы по пустым символам, необходимо выбрать пункт меню *Test/Highlight ϵ -transitions*. В результате будет открыта новая закладка, в которой будут подсвечены переходы по пустому символу.