|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования**  **«Московский государственный технический университет имени Н. Э. Баумана**  **(национальный исследовательский университет)» (МГТУ им. Н. Э. Баумана)** |

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

**Лабораторная работа № 1**

**Дисциплина:** Конструирование компиляторов

**Студент:** Платонова Ольга

**Вариант:** 4

**Группа:** ИУ7-22M

**Преподаватель:** Ступников А. А.

Москва, 2023 г.

**Цель работы:** приобретение практических навыков реализации важнейших элементов лексических анализаторов на примере распознавания цепочек регулярного языка.

**Задачи работы:**

1) Ознакомиться с основными понятиями и определениями, лежащими в основе построения лексических анализаторов.

2) Прояснить связь между регулярным множеством, регулярным выражением, праволинейным языком, конечно-автоматным языком и недетерминированным конечно-автоматным языком.

3) Разработать, тестировать и отладить программу распознавания цепочек регулярного или праволинейного языка в соответствии с предложенным вариантом грамматики.

**Теоретическая часть**

*Недетерминированный конечный автомат (НКА)* состоит из:

1. множества состояний S;
2. множества входных символов Σ (входного алфавита); считаем, что символ , обозначающий пустую строку, не является членом Σ;
3. функции переходов, которая для каждого состояния и каждого символа из Σ ∪ {} дает множество последующих состояний (next state);
4. состояния s0 из S, известного как стартовое (начальное);
5. множества состояний F, являющегося подмножеством S, известных как допускающие (конечные).

*Детерминированный конечный автомат (ДКА)* представляет собой частный

случай НКА, в котором:

1. нет переходов для входа ;
2. для каждого состояния s и входного символа a имеется ровно одна дуга, выходящая из s и помеченная a.

*Алгоритм Томпсона* строит по НКА эквивалентный ДКА следующим образом:

Начало.

Шаг 1. Помещаем в очередь Q множество, состоящее только из стартовой вершины.

Шаг 2. Затем, пока очередь не пуста выполняем следующие действия:

* Достаем из очереди множество, назовем его q.
* Для всех c ∈ Σ посмотрим в какое состояние ведет переход по символу c из каждого состояния в q. Полученное множество состояний положим в очередь Q только если оно не лежало там раньше. Каждое такое множество в итоговом ДКА будет отдельной вершиной, в которую будут вести переходы по соответствующим символам.
* Если в множестве q хотя бы одна из вершин была терминальной в НКА, то соответствующая данному множеству вершина в ДКА также будет терминальной.

Конец.

*Алгоритм минимизации конечных автоматов Бржозовского*.

Введём следующие обозначения:

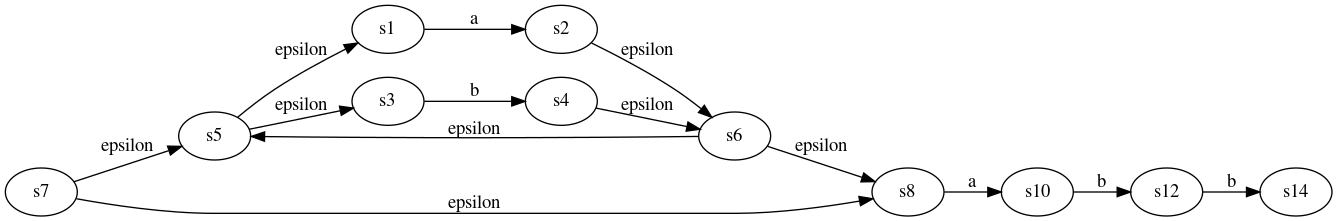
* A — конечный автомат,
* d(A) — детерминизированный автомат для A,
* r(A) — обратный автомат для A,
* dr(A) — результат d(r(A)). Аналогично для rdr(A) и drdr(A).

Пусть A — автомат (необязательно детерминированный), распознающий язык L. Минимальный детерминированный автомат AL может быть вычислен следующим образом: AL=drdr(A).

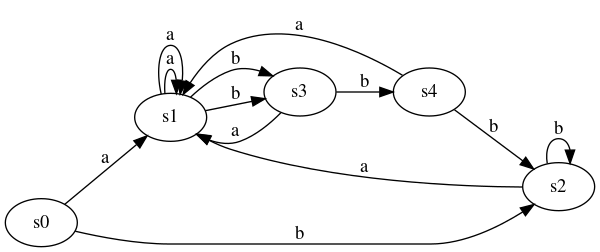
**Результаты работы**

Регулярное выражение: (a|b)\*abb

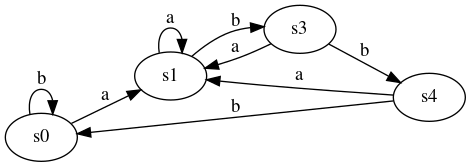
Недетерминированный конечный автомат:



Эквивалентный детерминированный конечный автомат:



Наименьший конечный автомат:



Входная цепочка: babb

Результат: True