

1 минимальный ДКА

Имеем регулярное выражение:

$$((a^*b^*c^*)^*ab(a^*b^*c^*)^*bc(a|b|c)^*) \mid ((a|b|c)^*bc(a^*b^*c^*)^*ab(a|bc|cc|bb)^*) \mid abc$$

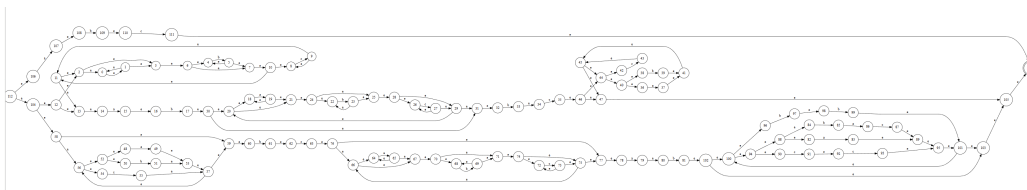
Преобразуем его в вид, удобный для парсинга. Явно обозначим конкатенацию символом «·»:

$$\begin{aligned} & ((a^* \cdot b^* \cdot c^*)^* \cdot a \cdot b \cdot (a^* \cdot b^* \cdot c^*)^* \cdot b \cdot c \cdot (a|b|c)^*) \mid \\ & ((a|b|c)^* \cdot b \cdot c \cdot (a^* \cdot b^* \cdot c^*)^* \cdot a \cdot b \cdot (a|b \cdot c|c \cdot c|b \cdot b)^*) \mid \\ & a \cdot b \cdot c \end{aligned}$$

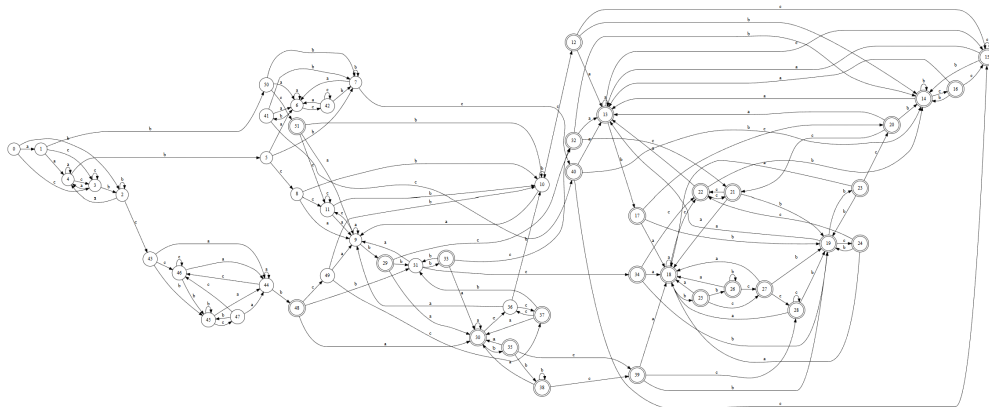
Преобразуем полученное выражение в Обратную Польскую Нотацию:

$$a * b * .c * . * a.b.a * b * .c * . * .b.c.ab|c| * .ab|c| * b.c.a * b * .c * . * .a.b.abc.|cc.|bb.| * .|ab.c.|$$

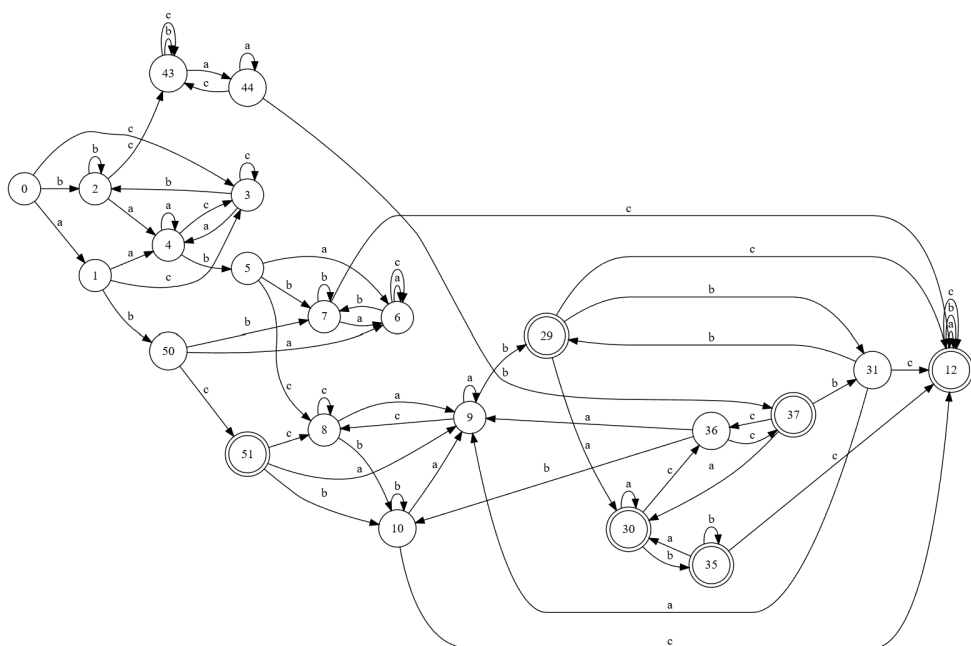
Применим метод Томпсона к обратной польской записи и НКА.



Выполним детерминизацию полученного НКА



А затем минимизацию полученного ДКА.



	e	a	b	c	ac	bc	cb	cc	bbc	cab	abc
e	-	-	-	-	-	-	-	-	-	-	+
a	-	-	-	-	-	+	-	-	-	-	-
b	-	-	-	-	-	-	-	-	-	+	-
c	-	-	-	-	-	-	-	-	-	-	-
aa	-	-	-	-	-	-	-	-	+	-	-
ab	-	-	-	+	-	+	-	-	+	+	+
bc	-	-	-	-	-	-	-	-	-	+	-
aab	-	-	-	-	-	+	-	-	+	+	+
aba	-	-	-	-	-	+	-	-	+	+	+
abb	-	-	-	+	+	+	+	+	+	+	+
abc	+	-	-	-	-	+	-	-	+	-	+
bca	-	-	+	-	-	-	+	-	+	+	-
aabc	-	-	-	-	-	+	-	-	+	+	+
abca	-	-	+	-	-	+	+	-	+	+	+
abcb	-	-	-	+	+	+	+	+	+	+	+
abbc	+	+	+	+	+	+	+	+	+	+	+
bcab	+	+	-	-	-	+	-	+	-	+	+
abcab	+	+	-	+	+	+	+	+	+	+	+
bcaba	+	+	-	-	-	+	-	-	+	+	+
bcabb	-	-	+	+	-	+	+	+	+	+	+
bcabc	-	-	-	+	-	+	-	-	+	+	+
bcabab	+	+	+	+	-	+	+	+	+	+	+

2 малый НКА

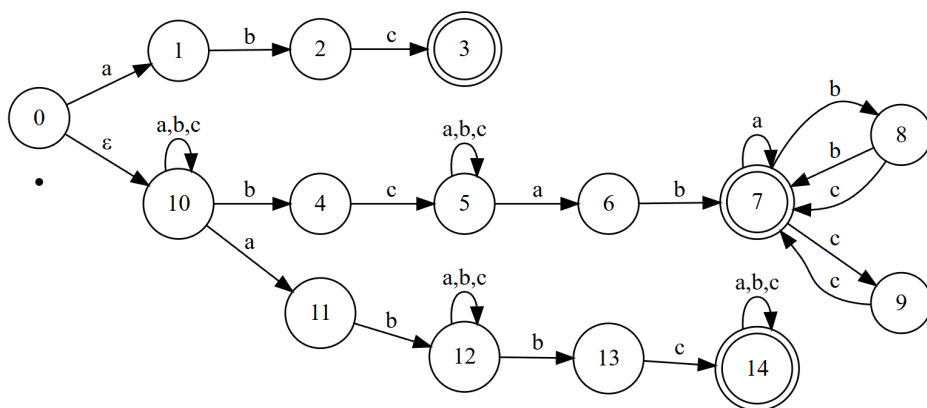
Упростим исходное регулярное выражение.

Заметим, что конструкция $(a^*b^*c^*)^*$ при условии алфавита $\Sigma = \{a, b, c\}$ эквивалентна $(a|b|c)^*$.

Упрощенное выражение принимает вид:

$$((a|b|c)^*ab(a|b|c)^*bc(a|b|c)^*) \mid ((a|b|c)^*bc(a|b|c)^*ab(a|b|c|cc|bb)^*) \mid abc$$

построим по нему малый НКА



Получилось выделить 7 различных КЭ. последний суффикс abbc никаким префиксом из языка не вывести и минус на нем не получить. Итак, в минимальном НКА как минимум 7 состояний

	e	b	c	ab	bc	cab	abbc
abc	+
bca	-	+
ab	-	-	+
bc	-	-	-	+	.	.	.
a	-	-	-	-	+	.	.
b	-	-	-	-	-	+	.
cc	-	-	-	-	-	-	+

3 Расширенные регулярные выражения

Вынесем проверку алфавита $\{a, b, c\}$ в lookahead. Тогда конструкции $(a|b|c)^*$ можно заменить на $.^*$.

$$^{\wedge} (? = [abc]^* \$) (.*ab.*bc.*|.*bc.*ab(a|bc|cc|bb)^*|abc) \$$$

4 ПКА

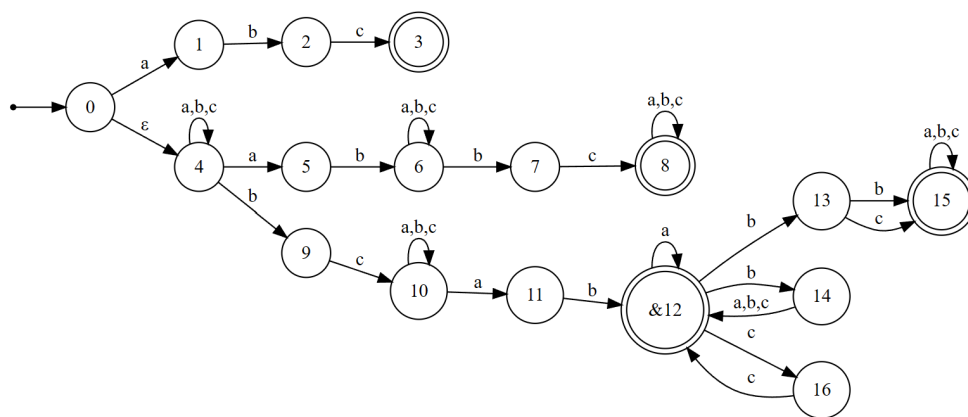
Наш язык L представляет собой объединение трех языков: $L = L_1 \cup L_2 \cup L_3$.

Соответственно И недетерминизм надо ввести в каком-нибудь L_i . У L_1 и L_3 структура линейна и проста, вводить распараллеливание негде

Остается L_2 . В нем префиксная часть $.^*bc.*ab$ также линейна. Единственное место, допускающее параллельную обработку — это суффикс: $(a | bc | cc | bb)^*$

Нас интересуют составные под слова bc и bb . Можно разделить обработку на два потока:

1. Поток 1: После нахождения b просто продолжает чтение, ожидая новые символы согласно структуре.
2. Поток 2: Проверяет, является ли текущая последовательность допустимым под словом (bc или bb) в контексте суффикса.



Нижняя оценка числа состояний ПКА - 3.

	a	bc	cab
ab	-	+	+
b	.	-	+
a	.	.	-