Формальные языки

домашнее задание до 23:59 05.03

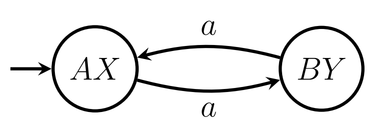
Лупуляк Василий, 18Б09

1. Доказать или опровергнуть утверждение: произведение двух минимальных автоматов всегда дает минимальный автомат (рассмотреть случаи для пересечения, объединения и разности языков).

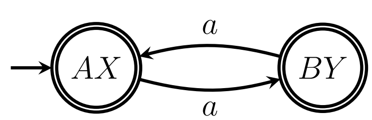
   Решение: для разности это утверждение неверно для равных автоматов из хотя бы двух вершин (так как разность будет приниматься пустым автоматом, а в произведении вершин будет хотя бы две). Для пересечения и объединения рассмотрим следующий контрпример: первый автомат принимает язык $\{a^n \mid n \,\vdots\, 2\}$, второй принимает $\{a^n \mid n \not\vdots\, 2\}$:

   Тогда после удаления недостижимых вершин произведение имеет вид

   или

   (для случаев пересечения и объединения соответственно). В обоих случаях при удалении состояния BY принимаемый язык не меняется, поэтому ни один из автоматов не является минимальным.

2. Для регулярного выражения:

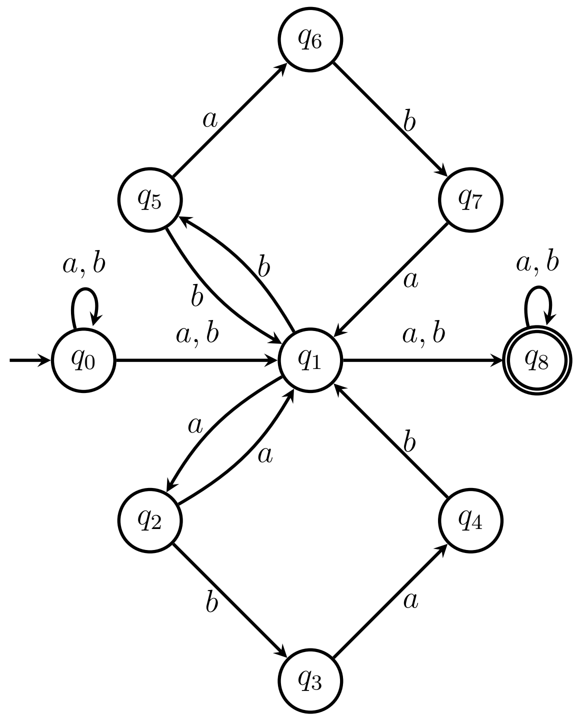
$$(a \mid b)^+(aa \mid bb \mid abab \mid baba)^*(a \mid b)^+$$

   Построить эквивалентные:

   (a) Недетерминированный конечный автомат

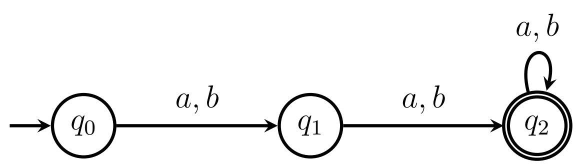
(b) Недетерминированный конечный автомат без ε-переходов

Решение:

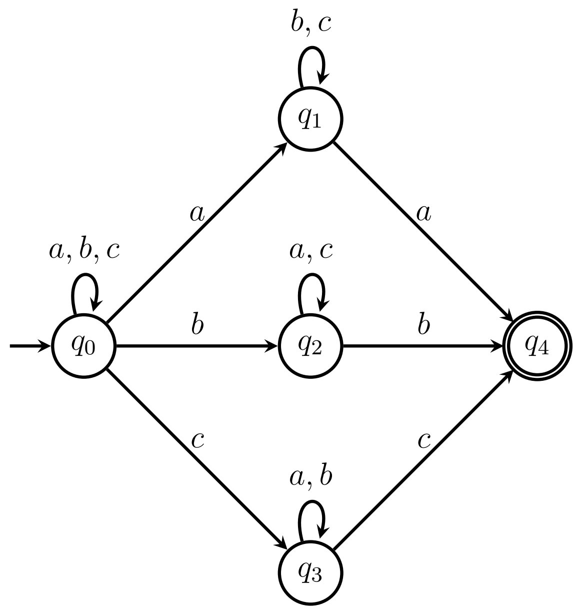


(c) Минимальный полный детерминированный конечный автомат

Решение: заметим, что все подходящие под выражение слова имеют длину не менее двух символов. С другой стороны, все такие слова подходят под выражение (т.к. имеют вид $(a \mid b)(a \mid b)^*(a \mid b)$, то есть подходят под $(a \mid b)^+\varepsilon(a \mid b)^+$). Значит, минимальный автомат выглядит следующим образом:



3. Построить регулярное выражение, распознающее тот же язык, что и автомат:



Решение: если автомат распознает некоторое слово ωx, $\omega \in \{a, b, c\}^*$, $x \in \{a, b, c\}$, то $|\omega|_x \geqslant 1$, так как выход из q_0 должен был быть совершен по символу x. Несложно заметить, что любое слово ωx с таким свойством принимается автоматом (выход из q_0 нужно произвести по последнему вхождению x в ω). Это свойство можно записать в виде регулярного выражения следующим образом:

$$[a\text{-}c]^*(a[a\text{-}c]^*a \mid b[a\text{-}c]^*b \mid c[a\text{-}c]^*c)$$

4. Определить, является ли автоматным язык $\{\omega\omega^r \mid \omega \in \{0, 1\}^*\}$. Если является — построить автомат, иначе — доказать.

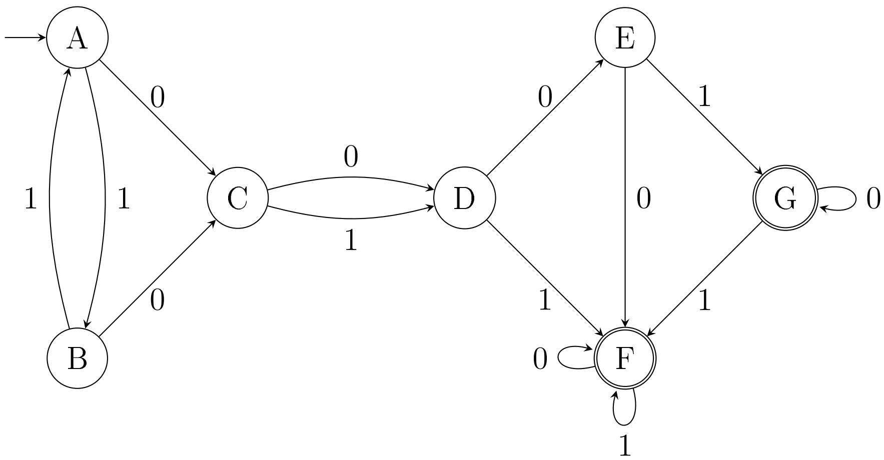
   Решение: предположим, что язык является автоматным. Тогда, по теореме о накачке, для достаточно большого N $\exists x, y, z : 1^N 001^N = xyz$, $|y| > 0$, $|xy| < N$ и $xyyz$ тоже лежит в языке. Но $xyyz = 1^{(N+|y|)}001^N$, то есть это слово не является палиндромом, что приводит к противоречию. Таким образом, язык не является автоматным.

5. Определить, является ли автоматным язык $\{uaav \mid u, v \in \{a, b\}^*, |u|_b \geq |v|_a\}$. Если является — построить автомат, иначе — доказать.

   Решение: предположим, что язык является автоматным. Применим теорему о накачке к слову $b^n aa(ba)^n = xyz$ (для n из условия теоремы). Тогда $y \in \{b\}^*$, так как $|xy| \leqslant n$. Тогда слово $xz = b^{(n-|y|)}aa(ba)^n$ должно лежать в языке, но оно не лежит, так как это слово единственным способом представляется в виде $uaav$ и при этом $|u|_b = n - |y| < n = |v|_a$. Получаем противоречие, тем самым доказав, что язык не является автоматным.

Пример применения алгоритма минимизации

Минимизируем данный автомат:



Автомат полный, в нем нет недостижимых вершин — продолжаем.

Строим обратное δ отображение.

δ^{-1}	0	1
A	—	B
B	—	A
C	A B	—
D	C	C
E	D	—
F	E F	D F G
G	G	E

Отмечаем в таблице и добавляем в очередь пары состояний, различаемых словом ε: все пары, один элемент которых — терминальное состояние, а второй — не терминальное состояние. Для данного автомата это пары

$(A, F), (B, F), (C, F), (D, F), (E, F), (A, G), (B, G), (C, G), (D, G), (E, G)$

Дальше итерируем процесс определения неэквивалентных состояний, пока очередь не оказывается пуста.

(A, F) не дает нам новых неэквивалентных пар. Для (B, F) находится 2 пары: $(A, D), (A, G)$. Первая пара не отмечена в таблице — отмечаем и добавляем в очередь. Вторая пара уже отмечена в таблице, значит, ничего делать не надо. Переходим к следующей паре из очереди. Итерируем дальше, пока очередь не опустошится.

Результирующая таблица (заполнен только треугольник, потому что остальное симметрично) и порядок добавления пар в очередь.

	A	B	C	D	E	F	G
A							
B							
C	✓	✓					
D	✓	✓	✓				
E	✓	✓	✓	✓			
F	✓	✓	✓	✓	✓		
G	✓	✓	✓	✓	✓		

Очередь:
$(A, F), (B, F), (C, F), (D, F), (E, F), (A, G), (B, G), (C, G), (D, G), (E, G),$
$(B, D), (A, D), (A, E), (B, E), (C, E), (C, D), (D, E), (A, C), (B, C))$

В таблице выделились классы эквивалентных вершин: $\{A, B\}, \{C\}, \{D\}, \{E\}, \{F, G\}$. Остается только нарисовать результирующий автомат с вершинами-классами. Переходы добавляются тогда, когда из какого-нибудь состояния первого класса есть переход в какое-нибудь состояние второго класса. Минимизированный автомат: