

Teoria Języków Formalnych

1. Języki regularne. Automaty skończone (deterministyczne i niedeterministyczne) i wyrażenia regularne oraz ich zastosowania

Języki regularny – język, dla którego istnieje deterministyczny automat rozstrzygający, czy dane słowo należy do tego języka.

Równoważnie: język, dla którego istnieje gramatyka regularna.

Przykłady języków regularnych nad alfabetem {a, b}:

- zbiór wszystkich słów alfabetu
- zbiór wszystkich słów alfabetu długości n
- zbiór wszystkich słów zaczynających się na aaabbb i kończących się na ababab

Automaty skończone

Automat to abstrakcyjna maszyna, która czyta słowo wejściowe i zmienia swój stan. Są dwa stany specjalne: początkowy (może być tylko 1) i akceptujący (może być wiele takich stanów).

Przed przeczytaniem pierwszej litery słowa wejściowego, automat znajduje się w stanie początkowym. Następnie automat czyta pierwszą literę i zmienia swój stan zgodnie z funkcją przejścia. Jeżeli po przeczytaniu całego słowa automat znajduje się w stanie

- akceptującym – słowo zostało zaakceptowane
- innym niż akceptujący – słowo nie zostało zaakceptowane

Automat rozpoznaje język L, jeżeli akceptuje każde słowo z tego języka.

Język L jest rozpoznawalny, jeżeli istnieje skończony automat, który go rozpoznaje.

Automat jest zupełny, jeżeli funkcja przejścia jest określona dla każdego stanu i litery.

Automat, który nie jest zupełny można zamienić na równoważny zupełny przez dodanie nieakceptującego stanu (pułapki) i dodanie wszystkich brakujących przejść tak, aby prowadziły do pułapki.

Automaty skończone deterministyczne i niedeterministyczne

W automacie deterministycznym dla dowolnego stanu i dowolnej litery istnieje co najwyżej jedna możliwość przejścia do innego stanu.

W automacie niedeterministycznym po przeczytaniu litery w jakimś stanie istnieje wiele możliwości zmiany stanu.

Każdemu niedeterministycznemu automatowi skończonemu odpowiada równoważny mu deterministyczny automat skończony.

Maszyna Turinga jest generalizacją automatu skończonego operującego na nieskończonej pamięci.

Wyrażenia regularne

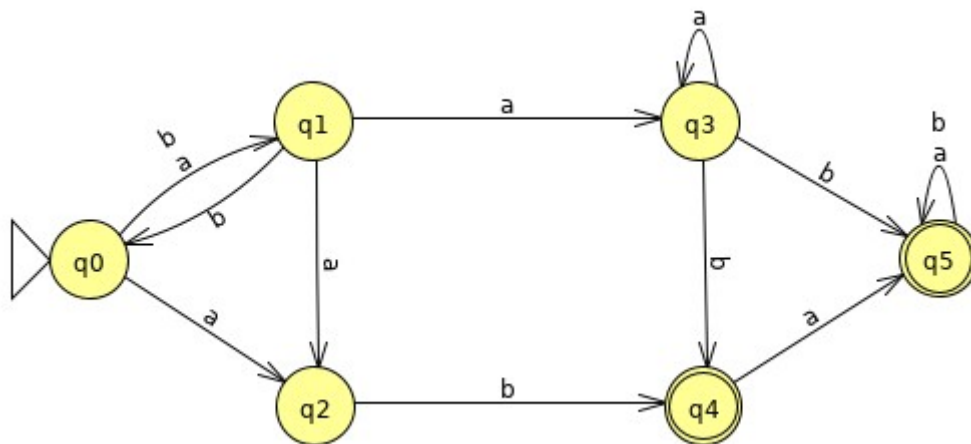
Wyrażenie regularne to wzorec, który opisuje jakiś język regularny. Dla każdego wyrażenia regularnego istnieje równoważny mu skończony automat deterministyczny.

Przykłady:

- $[0-9]\{2\}-[0-9]\{3\}$ – kod pocztowy
- $a..z$ – czteroznakowe słowo zaczynające się na a i kończące na z
- $(ala|ma|kota)$ – dopasowanie jednego z trzech słów
- $[a-f0-9]\{8\}-[a-f0-9]\{4\}-[a-f0-9]\{4\}-[a-f0-9]\{4\}-[a-f0-9]\{12\}$ – prawidłowy UUID
- $.*(?:.*=.*))$ - [regex, który zabił Cloudflare](#)

Przykłady automatów (nad alfabetem {a, b})

Przykład 1



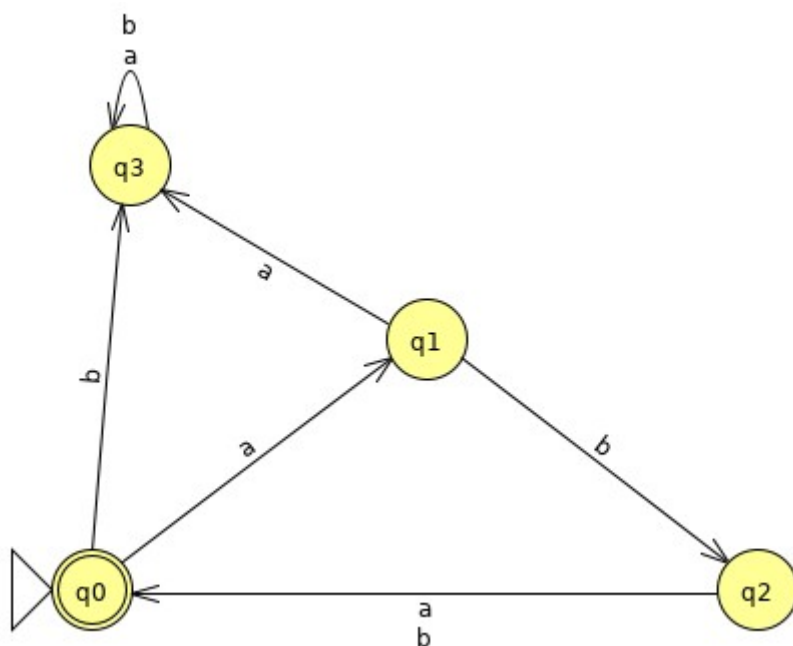
Automat jest niedeterministyczny, ponieważ:

- w stanie q0 po przeczytaniu a można przejść do q1 lub q2
- w stanie q1 po przeczytaniu a można przejść do q2 lub q3
- w stanie q3 po przeczytaniu b można przejść do q5 lub q4

Automat nie jest zupełny, ponieważ funkcja przejścia nie jest określona dla

- stanu q2 i litery a
- stanu q4 i litery b

Przykład 2



Automat jest

- deterministyczny – nie ma stanu, z którego wychodziłaby więcej niż jedna strzałka z tą samą literą
- zupełny, bo dla każdego stanu i litery określona jest funkcja przejścia (strzałka)