

Matematyczna wieża Babel.

2. Automaty – o językach regularnych

materiały do ćwiczeń

Projekt „Matematyka dla ciekawych świata”
spisał: Michał Korch

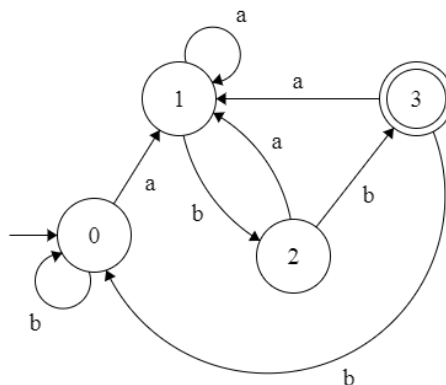
7 marca 2019

1 Automaty!

O automacie można myśleć jak na dwa zasadnicze równoważne sposoby. Możecie myśleć o nim jako o maszynie, która ma pewną (skończoną) liczbę stanów. Jest wśród tych stanów stan początkowy, w którym maszyna znajduje się przed rozpoczęciem pracy. Automat taki będzie dostawać (czy odczytywać) pewne słowo i w zależności od tego przechodzić do innych stanów. Precyzyjnie mówiąc nowy stan maszyny zależy od tego w jakim stanie maszyna się znajduje i jaka litera została wczytana. Maszyna ma za zadanie zaakceptować lub odrzucić wczytywane słowo – dlatego wśród stanów jest jeden lub więcej stanów akceptujących. Jeśli po przeczytaniu słowa maszyna znajduje się w jednym z nich, to słowo jest akceptowane. Jeśli jest w dowolnym innym stanie, słowo jest odrzucane.

Równoważnie, możecie myśleć o automacie jako o grafie, którego wierzchołki to stany (i jest jeden wierzchołek początkowy oraz pewne wierzchołki akceptujące), a krawędzie to strzałki pomiędzy tymi wierzchołkami podpisane poszczególnymi literami. Wtedy działanie automatu to pewna droga przez graf zaczynająca się w wierzchołku początkowym i każdorazowa idąca z wierzchołka do wierzchołka strzałką podpisaną daną literą.

Na przykład to jest automat:



Stan początkowy jest oznaczony dodatkową strzałką, zaś stan końcowy podwójnym kółkiem.

Zadanie 1

Prześledźcie działanie automatu dla następujących słów: *aba*, *abb*, *aababb*, *abbab*, *abababa*. Jakie są kolejne stany po wczytaniu kolejnych liter przez automat? Które z tych słów są przez niego akceptowane?

Zadanie 2

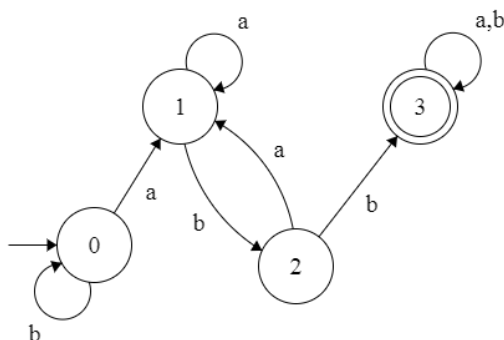
Jaka jest cecha wyróżniająca wszystkie słowa akceptowane przez ten automat, od słów, które nie są przez niego akceptowane? Udowodnijcie, że teza, którą stawiacie jest prawdziwa.

Wskazówka: Znajdźcie wspólną cechę akceptowanych słów. Następnie trzeba wykazać, że tylko słowa mające tę cechę są akceptowane oraz wykazać, że żadne słowo mające tę cechę nie może być odrzucone.

Zbiór wszystkich słów, które będą przez automat zaakceptowane nazywa się językiem rozpoznawanym przez ten automat.

Zadanie 3

Zmodyfikujmy nieco rozpatrywany automat

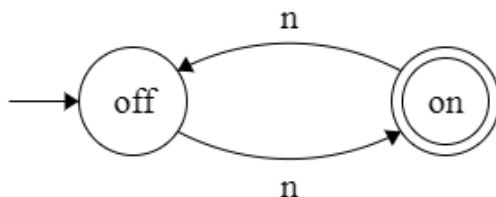


Jaki tym razem język jest przez ten automat rozpoznawany?

Wskazówka: Jaka jest różnica pomiędzy tym automatem a poprzednim? Co dzieje się, gdy dotrzemy do stanu akceptującego, ale jeszcze są jakieś litery do wczytania.

2 Automaty a życie

Wiele procesów zachodzącym w „prawdziwym świecie” da się opisać przy pomocy automatu. Najprostszym przykładem jest chociażby włącznik światła. Taki automat ma dwa stany: „on” i „off” (załóżmy, że stan początkowy to „off”). Ma tylko jeden znak, z którego mogą być złożone „słowa”, które czyta. Nazwijmy ten znak *n* (jak naciśnij). Pod wpływem *n* włącznik zmienia swój stan z „on” na „off” i na odwrót. Dla ustalenia uwagi, powiedzmy, że stan akceptujący to jest zapalone światło. A zatem mamy taki automat:



Oczywiście ten automat „akceptuje” (włącznik ma zapalone światło) słowa złożone z nieparzystej liczby n (nieparzystej liczby naciśnięć).

Zadanie 4

Znajdźcie jakąś inną rzecz z „prawdziwego świata”, którą da się opisać prostym automatem i to zróbcie!

3 Nie każdy język jest regularny!

Przypomnę, że zbiór słów nazywamy językiem regularnym, jeśli istnieje automat, który go rozpoznaje (czyli akceptuje słowo wtedy i tylko wtedy, gdy należy do tego języka). Na przykład, w pierwszej części zobaczyliśmy, że zbiór wszystkich słów kończących się na abb jest językiem regularnym. Podobnie zbiór wszystkich słów, które zawierają podśowo abb . Okazuje się jednak, że nie każdy język jest regularny.

Narzędziem pozwalającym łatwo dowieść, że jakiś język nie jest regularny jest tzw. „lemat o pompowaniu”. Porozmawiajmy o nim na konkretnym przykładzie. Okazuje się, że chociażby język wszystkich słów, które są postaci: n liter a , a potem n (czyli tyle samo) liter b nie jest regularny. Udowodnijmy to! Załóżmy przeciwnie, że ten język jest regularny, czyli istnieje automat, który go rozpoznaje. W takim razie automat ten ma ileś stanów powiedzmy, że N . Weźmy zatem słowo, które najpierw ma $N + 1$ liter a , a potem $N + 1$ liter b . Ale automat ma tylko N stanów. To oznacza, że przechodząc przez pierwsze $N + 1$ liter a musi odwiedzić co najmniej jeden swój stan (nazwijmy ten stan S), co najmniej dwa razy. Jak spojrzymy na graf tego automatu, to znaczy, że czytając kolejne litery a przeszliśmy cyklem ze stanu S znów do tego samego stanu S . Powiedzmy, że ten cykl wymagał on nas przeczytania k liter a . Na koniec jednak czytane słowo musi być zaakceptowane, a więc przeczytawszy resztę liter a oraz $N + 1$ liter b trafiamy na stan akceptujący.

Zastanówmy się jednak, co by było, gdyby nasz automat czytał słowo, w którym jest $N + k + 1$ liter a , a potem $N + 1$ liter b . Początek byłby taki sam – dojdziemy do stanu S , a potem zrobimy cykl i znajdziemy się znów w stanie S – jak poprzednio. Teraz jednak przed nami kolejne k liter a . Ale wiemy, że po przeczytaniu kolejnych k liter a ze stanu S znów znajdziemy się w stanie S . No dobrze, a dalej będzie znów tak jak poprzednio – przeczytanie reszty liter a oraz $N + 1$ liter b przeprowadzi nas ze stanu S do stanu akceptującego. Czyli automat zaakceptował też i to słowo. A nie powinien! Bo ma akceptować tylko słowa postaci n liter a , a potem n liter b . Sprzeczność!

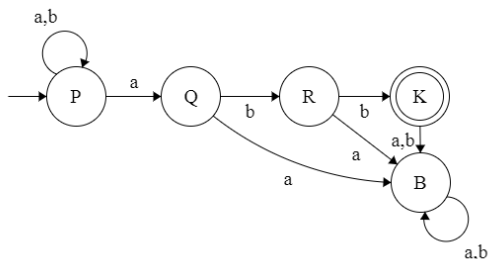
Zadanie 5

Udowodnijcie w podobny sposób, że język wszystkich palindromów (czyli słów takich, które wyglądają tak samo, gdy czytać je od początku i od końca) nie jest regularny.

Wskazówka: Podobnie jak poprzednio, załóżmy że ten język jest jednak regularny i w związku z tym istnieje automat, który go rozpoznaje. Niech ma on N stanów. Rozważmy słowo postaci $N + 1$ liter a , potem litera b , a potem $N + 1$ liter a . To jest palindrom, więc słowo to powinno być rozpoznane przez ten automat. Ale...

4 Automaty niedeterministyczne

Możemy trochę rozszerzyć definicję automatu dopuszczając więcej niż jeden stan początkowy oraz niejednoznaczność przejść (z jednego stanu może wychodzić wiele przejść podpisanych tą samą literą). Na przykład poniższy automat jest niedeterministyczny.

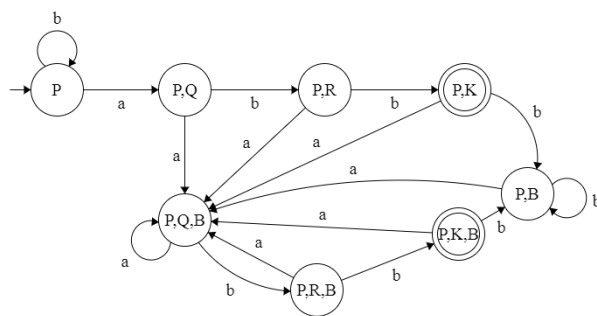


Jego niedeterminizm tkwi w tym co się dzieje w stanie początkowym po wczytaniu litery a . Są tutaj dwie opcje. Automat może pozostać w stanie P lub przejść do stanu Q . Powiemy, że automat niedeterministyczny akceptuje pewne słowo, o ile **istnieje** taka trasa przez ten automat wynikająca z kolejnych liter tego słowa, która prowadzi do stanu akceptującego. Inaczej mówiąc, jeśli ze wszystkich możliwych biegów automatu pod wpływem tego słowa istnieje taki, który kończy się na stanie akceptującym.

Zadanie 6

Które słów: aba , abb , $aababb$, $abbab$, $abababa$ są akceptowane przez powyższy automat? W przypadku słów akceptowanych, jak wygląda bieg, który kończy się w stanie akceptującym? Jaki język jest rozpoznawany przez ten automat?

Okazuje się, że każdy automat niedeterministyczny można „przerobić” na automat deterministyczny. Oto „przeróbka” rozważanego wyżej automatu.



Zadanie 7

Jaki jest dokładnie związek tego automatu z automatem niedeterministycznym z poprzedniego rysunku. Czemu odpowiadają stany nowego automatu? Dlaczego przejścia w nim są takie, a nie inne?

Wskazówka: Popatrzcie na przykład na stan P,Q . Do jakich stanów można dojść przy pomocy litery a w oryginalnym automacie, jeśli jest się w stanie P lub w stanie Q ?

5 Wyrażenia regularne

Jest jedno specjalne słowo, nie zawierające żadnego znaku, czyli słowo puste. Będziemy je oznaczać ε .

Oznaczenie to będzie nam potrzebne czasem w wyrażeniach regularnych. Wyrażenie regularne to pewien ciąg znaków który koduje pewien język (czyli ciąg słów). Wyrażenia regularne konstruuje i rozumie się następująco:

- ε jest wyrażeniem regularnym i oznacza język złożony z tylko tego pustego słowa, czyli $\{\varepsilon\}$,
- pojedyncza litera, np. a jest wyrażeniem regularnym i oznacza język złożony tylko z pojedynczego słowa będącego tą literą.
- jeśli v i w są pewnymi wyrażeniami regularnymi, to $v.w$ też jest wyrażeniem regularnym (nazywanym konkatenacją tych dwóch wyrażen). Oznacza język wszystkich słów, które mogą być podzielone tak, że początek należy do języka opisanego przez v , a koniec do języka opisywanego przez w . Na przykład $a.b$ oznacza język złożony z jednego słowa $\{a, b\}$.
- jeśli v i w są pewnymi wyrażeniami regularnymi, to $v + w$ też jest wyrażeniem regularnym (nazywanym sumą tych dwóch wyrażen). Oznacza język wszystkich słów, które są w języku opisywanym przez v lub w języku opisywanym przez w . Zatem np. $a + \varepsilon$ oznacza język złożony z dwóch słów $\{\varepsilon, a\}$. Natomiast $a.(a+b)$ oznacza język złożony z dwóch słów $\{aa, ab\}$. Oczywiście, w wyrażeniach regularnych możemy używać nawiasów aby określić kolejność operacji. Język $(a.a) + b$ to język $\{aa, b\}$.
- jeśli v jest wyrażeniem regularnym to v^* jest też wyrażeniem regularnym (gwiazdka Kleenego, domknięcie v). Oznacza to język słów, będących wielokrotną (w dowolnej liczbie, też zero) konkatenacją słów z języka opisywanego przez v . Na przykład $a^* = \{\varepsilon, a, aa, aaa, \dots\}$, oraz $(a + b)^* = \{\varepsilon, a, b, aa, ab, bb, aaa, aab, aba, abb, baa, bab, bba, bbb, \dots\}$.

Zadanie 8

Opisz jakie języki oznaczają następujące wyrażenia regularne:

- $(a.b)^*$
- $(a + b)^*.a$
- $(a + b)^*.a.b.b.(a + b)^*$
- $(0^* + (1.0^*.1))^*$

Jednym z najważniejszych twierdzeń w teorii automatów jest Twierdzenie Kleenego. Mówi ono, że język jest regularny (czyli rozpoznawany przez pewien automat) wtedy i tylko wtedy, gdy da się go opisać pewnym wyrażeniem regularnym.

Spróbujmy teraz znaleźć automat (być może niedeterministyczny), który rozpoznaje język zadany pewnym wyrażeniem regularnym.

Zadanie 9

Znajdźcie automat rozpoznający język opisany wyrażeniem regularnym $(a.a.b + b.a)^*.a.a^*$.

Wskazówka: Najpierw zastanówcie się jak wygląda taki automat dla wyrażenia $a.a.b$?

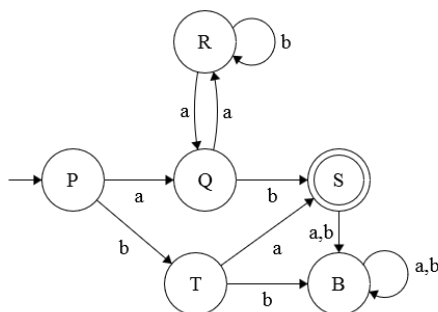
Wskazówka: Co trzeba dodać, by uzyskać $a.a.b + b.a$?

Wskazówka: To teraz zastanówcie się jak zmodyfikować ten automat, by dostać $(a.a.b + b.a)^*$.

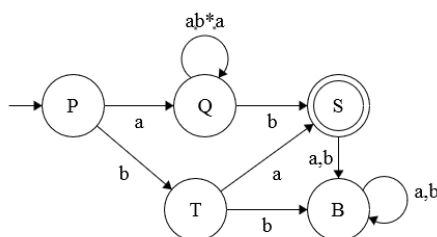
Wskazówka: Dorzućmy zatem końcówkę naszego wyrażenia $(a.a.b + b.a)^*.a.a^*$. Co wtedy?

Wskazówka: Na koniec, dla pełności możecie dodać stan „śmieciowy” i przejścia do niego ze stanów, z których do tej pory nie było kompletu przejść.

Pora zrobić to w drugą stronę. Znajdźmy wyrażenie regularne, które opisuje język opisywany przez dany automat. Cały trik polega na tym, żeby oprócz liter dopuścić w przejściach dowolne wyrażenia regularne (wtedy to przejście znaczy, że automat zmienia stan nie pod wpływem pojedynczej litery, ale pod wpływem słowa, które pasuje do wyrażenia regularnego). Dzięki temu można będzie usuwać kolejne stany automatu zmieniając przejścia na coraz bardziej skomplikowane wyrażenia regularne (żeby sam automat był równoważny), aby w końcu zostały tylko stan początkowy i końcowy, a łączyć je będzie przejście opisane szukanym wyrażeniem regularnym. Najłatwiej to zobaczyć na przykładzie. Zacznijmy od następującego automatu.



Zauważmy że stan R można usunąć dodając strzałkę ze stanu Q do Q opisaną wyrażeniem regularnym $a.b^*.a$, bo dokładnie takie słowa może wytworzyć wizyta w stanie R . Dostajemy więc



Zadanie 10

Spróbujcie kontynuować proces usuwania stanów, aby osiągnąć tylko dwa – początkowy i akceptujący połączone strzałką opisaną szukanym wyrażeniem.

Wskazówka: Można spokojnie usunąć stan B oraz przejścia do niego prowadzące, ponieważ z niego nie da się dojść do stanu akceptującego.

Wskazówka: Pora usunąć stan T zastępując go odpowiednim przejściem z P do S .

Wskazówka: Jakim przejściem możecie teraz zastąpić całe bebechy automatu, by zostawić tylko stan początkowy i akceptujący?

6 Logika monadyczna

Okazuje się również, że nie dość, że każdy język regularny można opisać automatem i można opisać wyrażeniem regularnym, to można też go opisać formułą logiczną. W tych formułach logicznych litery $x, y, z, \dots$ będą oznaczać pozycję w słowie. Więc $x = y$ oznacza, że x i y to ta sama pozycja, a $x < y$, że pozycja x jest w słowie przed pozycją y . Ponadto możemy używać sformułowania $\text{nast}(x, y)$, które oznacza, że pozycja y to kolejna pozycja w słowie (kolejna litera) po pozycji x . W formułach będą też występować zbiory pozycji $X, Y, Z, \dots$. Zatem $x \in X$ oznacza, że pozycja x należy do zbioru X . Można również używać tzw. kwantyfikatorów:

- $\forall_x$ dla każdej pozycji w słowie $x, \dots$,
- $\forall_X$ dla każdego zbioru pozycji $X, \dots$,
- $\exists_x$ istnieje taka pozycja w słowie x , że $\dots$,
- $\exists_X$ istnieje taki zbiór pozycji $X, \dots$.

Dla każdej litery, np. dla litery a , $a(x)$ będzie oznaczać, że na pozycji x w słowie jest litera a .

A zatem na przykład, $\forall_x a(x)$ jest formułą, która mówi, że wszystkie litery w słowie to a . Natomiast $\exists_x \exists_y x < y \wedge a(x) \wedge b(y)$ oznacza, że w słowie są dwie pozycje takie, że na pierwszej z nich stoi a , a na drugiej b . Na przykład słowo $aaab$ spełnia tę formułę. Natomiast słowo $baaaac$ jej nie spełnia. Językiem definiowanym przez formułę nazywamy zbiór wszystkich słów, które ją spełniają.

Zadanie 11

Wypiszcie przykłady dwóch słów spełniających i nie spełniających poniższych formuł. Opiszcie języki zdefiniowane przez te formuły.

- a) $\exists_x \exists_y a(x) \wedge b(y) \wedge \text{nast}(x, y)$,
- b) $\exists_X (\forall_x ((x \in X \leftrightarrow (b(x) \vee c(x)))) \wedge \neg \exists_y \exists_x \exists_z (x \in X \wedge (\neg y \in X) \wedge z \in X \wedge x < y < z))$.
Wskazówka: Co o X mówi formuła $x \in X \leftrightarrow (b(x) \vee c(x))$?
Wskazówka: Co o X mówi formuła $\neg \exists_y \exists_x \exists_z (x \in X \wedge (\neg y \in X) \wedge z \in X \wedge x < y < z)$?

Zadanie 12

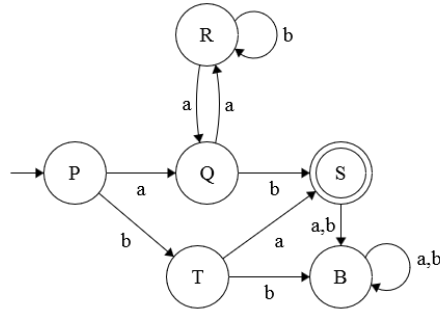
Opiszcie formułą język akceptowany przez automat, który rozważaliśmy w zadaniu 1.

Wskazówka: Zastanówcie się jak powiedzieć w logice monadycznej, że jakaś pozycja jest na końcu słowa.

7 Zadania dodatkowe

Zadanie 13

Rozważmy następujący automat



Prześledźcie działanie automatu dla następujących słów: ba , ε , $aabbbab$, $baaabbbab$. Jakie są kolejne stany po wczytaniu kolejnych liter przez automat? Które z tych słów są przez niego akceptowane?

Zadanie 14

Opiszcie język, który jest rozpoznawany przez ten automat.

Zadanie 15

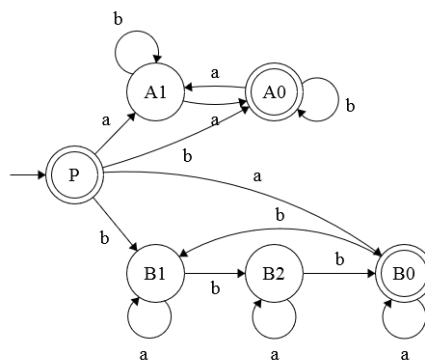
Językiem wyrażeń arytmetycznych nazywamy język nad alfabetem

$$\{ (,), 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, +, -, / \}$$

ciągów znaków będących poprawnymi wyrażeniami arytmetycznymi. Udowodnij, że nie jest to język regularny.

Zadanie 16

Rozważmy poniższy automat niedeterministyczny.



Które słów: aba , abb , $aababb$, $abbab$, $abababa$ są akceptowane przez powyższy automat? W przypadku słów akceptowanych, jak wygląda bieg, który kończy się w stanie akceptującym? Jaki język jest rozpoznawany przez ten automat?

Zadanie 17

Przekształćcie rozważany wyżej automat niedeterministyczny na deterministyczny.

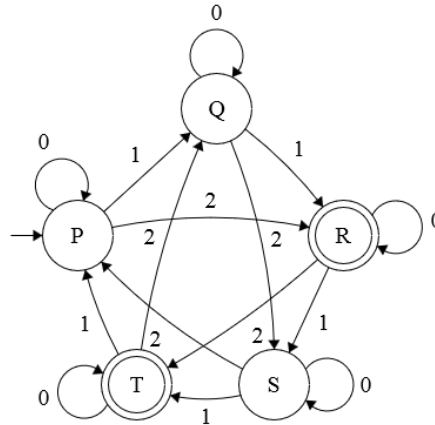
Zadanie 18

Znajdźcie wyrażenie regularne opisujące język rozpoznawany przez automat z zadania 16 (podobnie jak w zadaniu 9).

8 Praca domowa

Zadanie 19 (1 punkt)

Rozważ następujący automat czytający słowa złożone z cyfr 0, 1, 2.



Jakie kolejno stany będą odwiedzane pod wpływem słów: 010201, 120, 2221, 12010? Które z nich będą zaakceptowane?

Zadanie 20 (2 punkty)

Jaki język rozpoznaje rozpatrywany w poprzednim zadaniu automat?

Zadanie 21 (3 punkty)

Zmodyfikuj tak rozpatrywany automat, aby otrzymać automat niedeterministyczny, który akceptuje tylko takie słowa, które są akceptowane przez oryginalny automat, ale kończą się dwoma cyframi 1. Następnie zaproponuj automat deterministyczny rozpoznający identyczny język.

Wskazówka: W pierwszym przypadku wystarczy dodać 3 stany (wliczając stan kodujący błąd).

Wskazówka: Rysując drugi automat użyj trzech różnych kolorów do przejść opisanych 0, 1 i 2, aby zwiększyć czytelność rozwiązania.