Języki formalne i automaty

kolokwium 1 (5.12.2023)

1. Skonstruuj graf skończonego automatu deterministycznego akceptującego język

$$L = \{a^m b^n : \quad mn \ \text{dzieli się przez } 3\}.$$

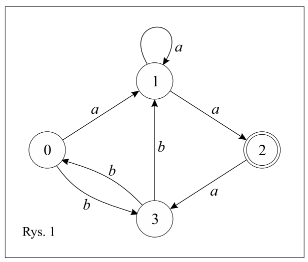
2. Zbuduj wyrażenie regularne równoważne automatowi niedeterministycznemu z Rys. 1.

3. Podaj opis automatu deterministycznego równoważnego automatowi z Rys. 2.

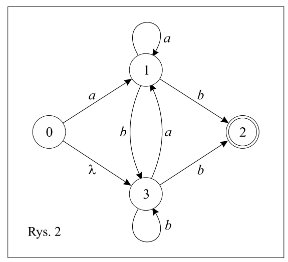
4. Napisz gramatykę regularną dla języka opisanego wyrażeniem $(a + ab)(aa + bab)^*bb + bab$.

5. Udowodnij, że język $L = \{\omega : |\omega|_a \leq |\omega|_b \leq 2|\omega|_a\}$ nie jest regularny.

6. **(Bonus)** Narysuj graf skończonego automatu deterministycznego M akceptującego te ciągi zero-jedynkowe, w których liczba jedynek na pozycjach nieparzystych jest nieparzysta, a na parzystych — parzysta (np.: 1, 1101, 11111, 011000111).



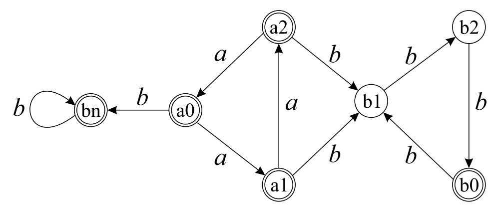
Rys. 1



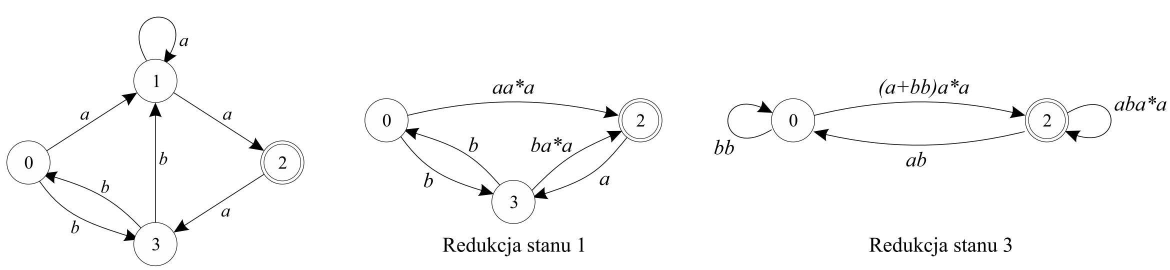
Rys. 2

ROZWIĄZANIA

1. Zauważmy, że mn dzieli się przez 3 jedynie gdy m lub n jest podzielne przez 3. Stan startowy automatu akceptującego L to $a0$. Stany $a1$ i $a2$ rejestrują liczbę m rozpoznanych liter a mod 3. Jeśli m dzieli się przez 3, liczba n liter b może być dowolna: takie słowa akceptowane w stanach końcowych $a0$ lub bn. Jeśli m nie dzieli się przez 3, wówczas akceptujemy słowa, w których liczba n liter b wynosi 0 (stany $a1$ i $a2$ są więc końcowe) lub jest dodatnią wielokrotnością 3 (stan końcowy $b0$).



2. Kolejne kroki redukcji automatu przedstawia rysunek:



Redukcja stanu 1

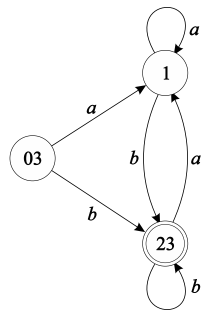
Redukcja stanu 3

Stąd odczytujemy postać równoważnego automatowi wyrażenia regularnego jako

$$(bb)^*(a + bb)a^*a\big(aba^*a + ab(bb)^*(a + bb)a^*a\big)^*.$$

3. Przejścia automatu niedeterministycznego zebrane są w tabelce. Automat deterministyczny zbudowany na jej podstawie przedstawiony jest na rysunku. Nowy stan początkowy to $\{0, 3\}$, ponieważ automat niedeterministyczny może rozpocząć działanie zmieniając stan z 0 na 3 przez puste przejście.

Stan	a	b
0	1	$2, 3$
1	1	$2, 3$
2	Ø	Ø
3	1	$2, 3$



4. Gramatyka regularna (prawostronnie liniowa)

$$\begin{array}{lcl} S & \to & aX \mid abX \mid bab \\ X & \to & aaX \mid babX \mid bb \end{array}$$

5. Załóżmy, że L jest regularny i niech m będzie stałą z Lematu o pompowaniu. Weźmy słowo

$$\alpha = a^m b^m .$$

W każdym podziale tego słowa $\alpha = \beta\,\gamma\,\delta$, spełaniającym warunki Lematu, $|\beta\gamma| \leq m$, część $\beta\gamma$ zawiera wyłącznie litery a, a więc $\gamma = a^r$ dla pewnego $r \geq 1$. Pompowanie $\alpha_k = \beta\gamma^k\delta$ produkuje więc słowa postaci

$$\alpha_k = a^{m+kr} b^m ,$$

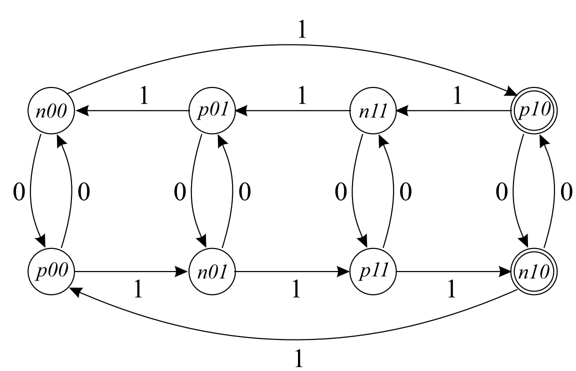
które dla $k \geq 1$ nie należą do L. Stąd L nie może być językiem regularnym.

Inaczej, korzystając z tw. Myhilla-Nerode'a określamy języki

$$L_k \;=\; \frac{dL}{da^k} .$$

Pokażemy, że wszystkie języki L_k są różne, jest ich więc nieskończenie wiele. Weźmy $k \neq j$ i niech $k < j$. Wtedy słowo $b^k \in L_k$, bo $a^k b^k \in L$, ale $b^k \notin L_j$ ponieważ $a^j b^k \notin L$. Z drugiej strony słowo $b^{2j} \in L_j$, ale $b^{2j} \notin L_k$, ponieważ $a^k b^{2j} \notin L$ wobec $2j > 2k$.

6. Proste rozwiązanie posiada 8 stanów etykietowanych przez (Wxy), gdzie W oznacza czytaną aktualnie pozycję ciągu wejściowego, nieparzystą $W = n$ lub parzystą $W = p$, natomiast x i y oznaczają przeczytane dotąd liczby jedynek odpowiednio na pozycjach nieparzystych i parzystych modulo 2. A więc np. $n01$ oznacza, że aktualnie czytamy kolejną nieparzystą pozycję w ciągu, przy czym liczba rozpoznanych dotąd jedynek na pozycjach nieparzystych była parzysta, a na parzystych — nieparzysta. Podobie $p11$ oznacza stan, w którym kolejna czytana pozycja ma numer parzysty, a liczby napotkanych dotąd jedynek na pozycjach nieparzystych i parzystych były nieparzyste. Stan początkowy to oczywiście $n00$, a stany końcowe zgodnie z warunkami zadania to $n10$ i $p10$.



Stosując inną metodę zbudujemy 4-stanowy automat optymalny (równoważny automat otrzymamy także przez optymalizację automatu z poprzedniego rysunku).

Zauważmy, że zbiór wszystkich ciągów $\{0,1\}^*$ można rozłożyć na 4 rozłączne klasy S_{pp}, S_{pn}, S_{np} i S_{nn}, opisane w tabeli poniżej:

Indeks	Liczba 1 na pozycjach nieparzystych	Liczba 1 na pozycjach parzystych
pp	parzysta	parzysta
pn	parzysta	nieparzysta
np	nieparzysta	parzysta
nn	nieparzysta	nieparzysta

W szczególności S_{np} to dokładnie nasz język $L = L(M)$. Zauważmy także, że dla dowolnego $\alpha \in \{0,1\}^*$ pochodna lewostronna $\frac{dL}{d\alpha}$ jest zawsze jedną z powyższych klas. Np.

$$\frac{dL}{d010} = \{\beta : \quad 010\beta \in L = S_{np}\} \; = \; S_{nn} \, .$$

Twierdzenie Myhilla–Nerode'a potwierdza więc, że L jest regularny, są bowiem tylko 4 możliwe wartości tej pochodnej. Zbudujemy automat M korzystając z konstrukcji przedstawionej w dowodzie twierdzenia M-N (str. 70–71).

Stany odpowiadają 4 możliwym wartościom pochodnej, dlatego użyjemy dla nich indeksów pp, np, pn i nn jako etykiet. Stanem startowym jest stan odpowiadający $\frac{dL}{d\lambda} = L = S_{np}$, oznaczony jako np. Stany końcowe odpowiadają zbiorom $\frac{dL}{d\alpha}$ zawierącym słowo λ, a więc tym, dla których $\alpha\lambda = \alpha \in L$. Ponieważ λ zawiera 0 jedynek na pozycjach nieparzystych i parzystych, należy ona do klasy S_{pp}. Zatem pp jest jedynym stanem końcowym. Funkcję przejścia otrzymamy analizując szczegółowo zbiory $\frac{dS_{pp}}{d0}$, $\frac{dS_{pp}}{d1}$, $\ldots$, $\frac{dS_{nn}}{d1}$. Wyniki tej analizy zawiera tabelka i graf poniżej:

Stan	0	1
pp	pp	pn
pn	np	nn
np	pn	pp
nn	nn	np