Języki, automaty i obliczenia

Wykład 6: Automaty na drzewach

Sławomir Lasota

Uniwersytet Warszawski

1 kwietnia 2015

Drzewa?

Pytanie

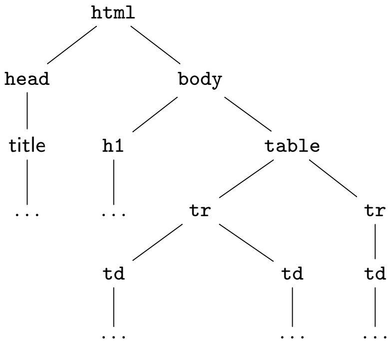
Dlaczego akurat drzewa?

Drzewa?

Pytanie

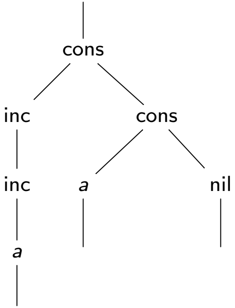
Dlaczego akurat drzewa?

```
<html>
    <head>
        <title> ...</title>
    </head>
    <body>
        <h1> ...</h1>
        <table>
            <tr>
                <td> ...</td>
                <td> ...</td>
            </tr>
            <tr>
                <td> ...</td>
            </tr>
        </table>
    </body>
</html>
```



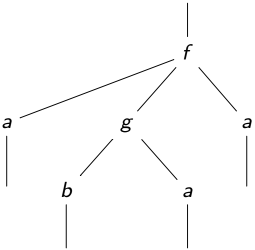
Drzewa – przykłady

$$A = \{a, \mathrm{nil}, \mathrm{inc}, \mathrm{cons}\}$$



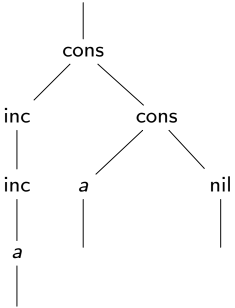
$$\begin{aligned}
\text{arność}(a) &= 1\\
\text{arność}(\mathrm{nil}) &= 1\\
\text{arność}(\mathrm{inc}) &= 1\\
\text{arność}(\mathrm{cons}) &= 2
\end{aligned}$$

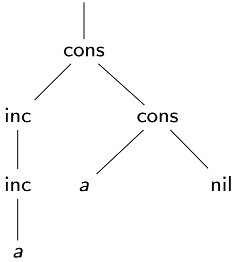
$$A = \{a, b, f, g\}$$



$$\begin{aligned}
\text{arność}(a) &= 1\\
\text{arność}(b) &= 1\\
\text{arność}(f) &= 3\\
\text{arność}(g) &= 2
\end{aligned}$$

Drzewa a termy





$\mathrm{cons}(\mathrm{inc}(\mathrm{inc}(a(\_))), \mathrm{cons}(a(\_), \mathrm{nil}(\_)))$

$\mathrm{cons}(\mathrm{inc}(\mathrm{inc}(a), \mathrm{cons}(a, \mathrm{nil}))$

Alfabet A z arnościami:

$$\text{arność} : A \to \mathbb{N} \setminus \{0\}$$

Drzewo nad A to para $t = (T, l)$, gdzie

$$T \subseteq \mathbb{N}^*, \qquad l : T \to A$$

t.że

- T jest zamknięty na prefiksy: $wv \in T \implies w \in T$
- jeśli $w \in T$ i $\text{arność}(l(w)) = n$, to

$$\{i \ : \ wi \in T\} \ = \ \{0 \dots n-1\} \qquad \text{albo} \qquad \{i \ : \ wi \in T\} \ = \ \emptyset$$

Notacja

W szczególnym przypadku, gdy

$$\text{arność}(a) \ = \ 1 \quad \text{dla wszystkich } a \in A,$$

otrzymujemy słowa nad A.

Bieg na słowie

słowo $a_1 a_2 \ldots a_n$:

a_1

a_2

$\ldots$

a_n

bieg akceptujący:

$\boxed{q_0}$ $\in I$

a_1

$\boxed{q_1}$

a_2

$\boxed{q_2}$

$\ldots$

$\boxed{q_{n-1}}$

a_n

$\boxed{q_n}$ $\in F$

Automaty na drzewach

$\mathcal{A} = (A, Q, I, F, \delta)$, gdzie

$$\delta \subseteq Q \times A \times Q^*$$

t.że

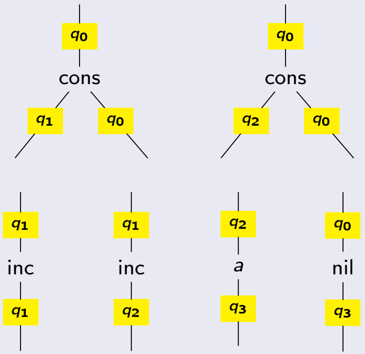
$$(q, a, w) \in \delta \implies |w| = \text{arność}(a)$$

Przykład

- $A = \{a, \text{nil}, \text{inc}, \text{cons}\}$

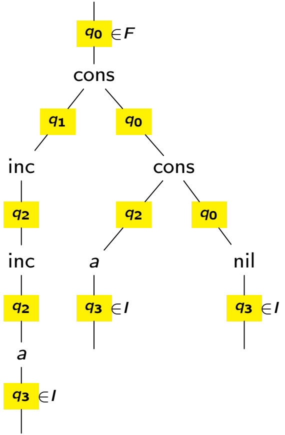
$$\begin{aligned} \text{arność}(a) &= 1 \\ \text{arność}(\text{nil}) &= 1 \\ \text{arność}(\text{inc}) &= 1 \\ \text{arność}(\text{cons}) &= 2 \end{aligned}$$

- $Q = \{q_0, q_1, q_2, q_3\}$
- $I = \{q_0\}$
- $F = \{q_3\}$
- $\delta = \{(q_0, \text{cons}, q_1 q_0), (q_1, \text{inc}, q_1),$
  $\{(q_0, \text{cons}, q_2 q_0), \ldots\}$

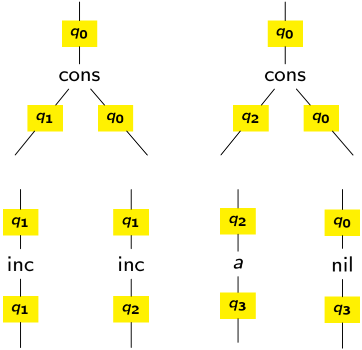


Bieg na drzewie – przykład

bieg akceptujący automatu:



przejścia automatu:



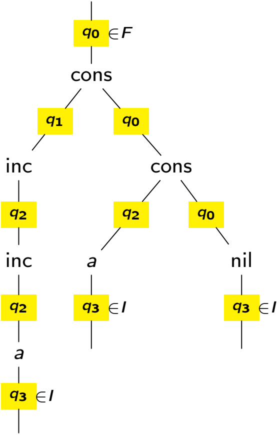
Co tu jest nie tak?

Bieg akceptujący automatu $\mathcal{A}$ na drzewie $t = (T, l)$ nad A to funkcja

$$f : T' \to Q, \qquad \text{gdzie } T' = T \cup \{wi \ : \ w \in T,\ 0 \leq i < \text{arność}(l(w))\}$$

t.że

- $(f(w), l(w), (f(wi))_{0 \leq i < \text{arność}(l(w))}) \in \delta$, dla każdego $w \in T$
- $f(\varepsilon) \in F$
- $f(w) \in I$, dla każdego $w \in T' - T$



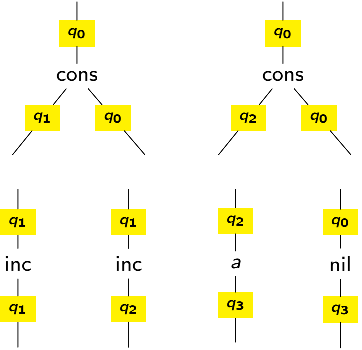
Deterministyczne automaty na drzewach

Automaty deterministyczne „z góry w dół”:

$$\forall q \in Q, a \in A.\ \exists! w \in Q^*.\ (q, a, w) \in \delta \qquad\qquad \delta_a : Q \to Q^{\text{arność}(a)}$$

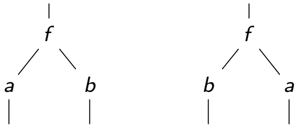
Automaty deterministyczne „z dołu do góry”:

$$\forall a \in A, w \in Q^{\text{arność}(a)}.\ \exists! q \in Q.\ (q, a, w) \in \delta \qquad\qquad \delta_a : Q^{\text{arność}(a)} \to Q$$



Determinizacja

Brak determinizacji „z góry w dół":

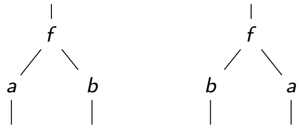


Pytanie

Czy zachodzi determinizacja „z dołu do góry" ?

Determinizacja

Brak determinizacji „z góry w dół":



Pytanie

Czy zachodzi determinizacja „z dołu do góry" ?

Odpowiedź

Konstrukcja podzbiorowa.

Pytanie

Które z poniższych języków są regularnymi językami drzew?

Które z nich są deterministyczne „z góry w dół” ?

$$A = \{a, b\}, \quad \text{arność}(a) = \text{arność}(b) = 2$$

- b występuje dokładnie raz
- b nie występuje ani raz
- na każdej ścieżce parzysta liczba a
- na pewnej ścieżce parzysta liczba a
- pod każdym a jest b

$$A = \{g, f, a, b, c\}, \quad \text{arność}(g) = 3, \text{arność}(f) = 2$$
$$\text{arność}(a) = \text{arność}(b) = \text{arność}(c) = 1$$

- $\{f(a^n, b^n) \ : \ n \geq 1\}$
- $\{g(c_1, g(c_2, g(\ldots g(c_n, c, c_n), \ldots), c_2), c_1) \ : \ c_1, \ldots, c_n \in \{a, b\}\}$

Konteksty

Pytanie

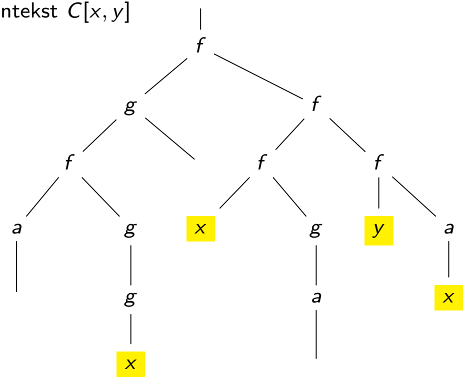
Czym jest „sufiks” drzewa? Czym jest „prefiks” drzewa? Czym jest „infiks” drzewa?

Konteksty

Pytanie

Czym jest „sufiks” drzewa? Czym jest „prefiks” drzewa? Czym jest „infiks” drzewa?

Kontekst $C[x, y]$



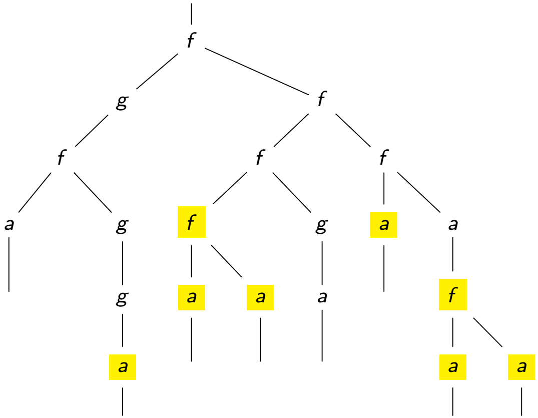
Kontekst trywialny $C[x]$

Co tu jest nie tak?

Kontekst jest *liniowy* jeśli każda zmienna występuje dokładnie raz.

Podstawienia

$C[f(a,a),a]$



Co tu jest nie tak?

Równoważność Myhilla-Nerode

Język L drzew nad A określa relację równoważności w zbiorze drzew nad A:

$$t \sim_L u \iff \forall C[x].\ (C[t] \in L \iff C[u] \in L)$$

Pytanie

Czy należy ograniczyć się do kontekstów liniowych $C[x]$?

Lemat o pompowaniu

Dla każdego regularnego języka drzew L istnieje n takie, że każde drzewo $t \in L$ o głębokości przynajmniej n można przedstawić jako

$$t = C[D[u]],$$

dla pewnych kontekstów liniowych $C[x], D[x]$ i dla pewnego drzewa u, takich, że

- kontekst $D[x]$ jest nietrywialny
- głębokość zmiennej w kontekście $C[D[X]]$ wynosi co najwyżej n,
- dla każdego $m \in \mathbb{N}$,

$$C[D^m[u]] \in L.$$

rysunek: pompowanie drzew

Pytanie

Czym jest homomorfizm z drzew nad A do drzew nad B ?

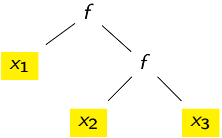
Homomorfizmy

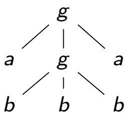
Pytanie

Czym jest homomorfizm z drzew nad A do drzew nad B ?

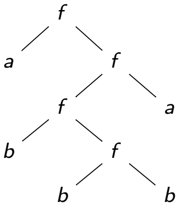
$$
\begin{aligned}
A &= \{g, a, b\} & \text{arność}(g) &= 3 & \text{arność}(a) &= \text{arność}(b) = 1 \\
B &= \{f, a, b\} & \text{arność}(f) &= 2
\end{aligned}
$$

$a \mapsto a$ $\quad$ $b \mapsto b$ $\quad$ $g \mapsto$





$\mapsto$



Operacje zachowujące regularność

Języki regularne drzew są zamknięte na następujące operacje:

- operacje boolowskie,
- obrazy homomorficzne,
- przeciwobrazy homomorficzne

Co tu jest nie tak?

Języki regularne drzew są zamknięte na następujące operacje:

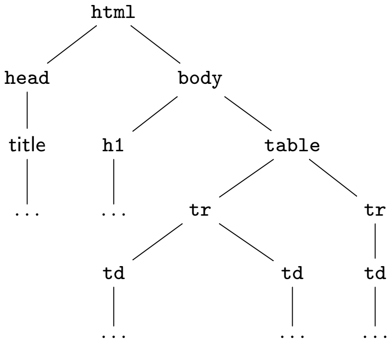
- operacje boolowskie,
- obrazy homomorficzne,
- przeciwobrazy homomorficzne

Co tu jest nie tak?

Pytanie

Czy należy ograniczyć się do homomorfizmów liniowych ?

```
<html>
    <head>
        <title> ...</title>
    </head>
    <body>
        <h1> ...</h1>
        <table>
            <tr>
                <td> ...</td>
                <td> ...</td>
            </tr>
            <tr>
                <td> ...</td>
            </tr>
        </table>
    </body>
</html>
```



W następnym odcinku:

języki bezkontekstowe