DEFINICE GRAMATIKY

GRAMATIKA G JE USPOŘÁDANÁ ČTVEŘICE

$$G = (N, T, S, P)$$

N ... MNOŽINA NETERMINÁLNÍCH SYMBOLŮ

T ... MNOŽINA TERMINÁLNÍCH SYMBOLŮ

$$N \cap T = \emptyset$$

$S \in N$ ... POČÁTEČNÍ SYMBOL

P ... MNOŽINA PŘEPISOVACÍCH PRAVIDEL

VE TVARU $\alpha \rightarrow \beta$

KDE $\alpha \in (N \cup T)^* N (N \cup T)^*$

$\beta \in (N \cup T)^*$

[NA LEVÉ STRANĚ PŘEPISOVACÍHO PRAVIDLA JE ALESPOŇ JEDEN NETERMINÁLNÍ SYMBOL, NA PRAVÉ MŮŽE BÝT COKOLI, I PRÁZDNÝ ŘETĚZEC.]

PŘ: PROGRAMOVACÍ JAZYK PASCAL

JE POPSÁN GRAMATIKOU REPREZENTOVANOU SYNTAKTICKÝMI DIAGRAMY.

POČÁTEČNÍ SYMBOL : PROGRAM

KONVENCE:

NETERMINÁLNÍ SYMBOLY:

<PODMĚT>, <IDENTIFIKÁTORY>

NEBO

$S, A, B, C, \ldots$

TERMINÁLNÍ SYMBOLY:

$0, 1, 2 \ldots$ NEBO

$a, b, c, d, f \ldots$

ŘETĚZCE:

$\alpha, \beta, \gamma, \delta \ldots$ NEBO

u, v, w, x, y, z

JAK GRAMATIKA DEFINUJE JAZYK ?

GRAMATIKA G DEFINUJE JAZYK L(G) „JAKO MNOŽINU VŠECH ŘETĚZCŮ, KTERÉ LZE V GRAMATICE ODVODIT"
[VERBÁLNÍ, MÁLO PŘESNÉ].

PŘESNĚJŠÍ VYJÁDŘENÍ:

NECHŤ $w \in (N \cup T)^* N (N \cup T)^*$

$z \in (N \cup T)^*$

ŘÍKÁME, ŽE w LZE PŘÍMO PŘEPSAT NA z PRÁVĚ TEHDY, KDYŽ EXISTUJÍ ŘETĚZCE $x_1, x_2, u, v \in (N \cup T)^*$ TAKOVÁ, ŽE

$w = x_1 u x_2 \quad z = x_1 v x_2 \quad u \to v \in P$

ZNAČENÍ: $w \Rightarrow z$

$$w \Rightarrow z \overset{def}{\iff} \exists u_1, u_2, u, v \in (N \cup T)^*: w = u_1 u u_2 \wedge z = u_1 v u_2 \wedge u \to v \in P$$

ŘÍKÁME, ŽE $\underline{w}$ LZE <u>PŘEPSAT</u> NA z PRÁVĚ TEHDY, KDYŽ EXISTUJE POSLOUPNOST ŘETĚZCŮ $w_0, w_1, \dots w_n \in (N \cup T)^*$ TAKOVÁ, ŽE

$$w = w_0 \Rightarrow w_1 \Rightarrow \dots \Rightarrow w_n = z$$

TATO POSLOUPNOST SE NAZÝVÁ ODVOZENÍM (DERIVACÍ) DÉLKY $\underline{n}$ SLOVA $\underline{z}$ ZE SLOVA $\underline{w}$.

ZNAČENÍ: $w \overset{*}{\Rightarrow} z$

PŘ: <PŘÍSUDEK> LZE PŘÍMO PŘEPSAT

NA BUYS

< $_{\text{PŘÍS. ČÁST}}$ > LZE PŘEPSAT

NA DRINKS GIN

PŘ:

<VĚTA> $\overset{*}{\Longrightarrow}$ OUR <PODMĚT> <PŘÍS. ČÁST>

α_1 μ α_2

$\Longrightarrow$

OUR JOHN <PŘÍS. ČÁST>

α_1 ν α_2

PROTOŽE <PODMĚT> $\rightarrow$ JOHN $\in P$

PŘI ODVOZOVÁNÍ ŘETĚZCE

„V DOSUD ODVOZENÉM ŘETĚZCI NAJDEME LEVOU STRANU NĚKTERÉHO PŘEPISOVACÍHO PRAVIDLA. ŘETĚZEC PŘEMÍSTÍME TAK, ŽE TUTO LEVOU STRANU NAHRADÍME ODPOVÍDAJÍCÍ PRAVOU STRANOU".

DEFINICE JAZYKA GENEROVANÉHO GRAMATIKOU

JAZYK GENEROVANÝ GRAMATIKOU JE MNOŽINA ŘETĚZCŮ TAKOVÝCH, ŽE

$$L(G) = \{ w \mid w \in T^* \wedge S \stackrel{*}{\Rightarrow} w \}$$

PŘ: $G = (N, T, S, P)$

$N = \{S\}$ $\quad P = \{ S \to 0S1,$

$T = \{0, 1\}$ $\quad S \to 01 \}$

ZJEDNODUŠENÝ ZÁPIS GRAMATIKY:

$S \to \overset{①}{0S1} \mid \overset{②}{01}$

ODVOZOVÁNÍ ŘETĚZCŮ:

$S \overset{②}{\Rightarrow} 01$

$S \overset{①}{\Rightarrow} 0S1 \overset{②}{\Rightarrow} 0011$

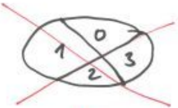
$S \overset{①}{\Rightarrow} 0S1 \overset{①}{\Rightarrow} 00S11 \overset{②}{\Rightarrow} 000111$

ZŘEJMĚ

$L(G) = \{ 0^n 1^n \; ; \; n \geq 1 \}$

CHOMSKÉHO KLASIFIKACE GRAMATIK

PODLE TVARU PŘEPISOVACÍCH PRAVIDEL



NE

ANO

POSTUPNĚ BUDEME „PŘITVRZOVAT" POŽADAVKY NA TVAR PRAVIDEL.

GRAMATIKA TYPU 0 (NEJOBECNĚJŠÍ)

$$\alpha \to \beta \qquad \alpha \in (N \cup T)^* N (N \cup T)^*$$

$$\beta \in (N \cup T)^*$$

[ODPOVÍDÁ NEJOBECNĚJŠÍ DEFINICI GRAMATIKY]

GRAMATIKA TYPU 1 (KONTEXTOVÁ) (NEVYPOUŠTĚCÍ)

$\alpha X \beta \rightarrow \alpha \gamma \beta$ $\alpha, \beta \in (N \cup T)^*$

$X \in N$

$\gamma \in (N \cup T)^+$

GRAMATIKA TYPU 2 (BEZKONTEXTOVÁ) (CFG)

$X \rightarrow \gamma$ $X \in N$

$\gamma \in (N \cup T)^*$

GRAMATIKA TYPU 3 (LINEÁRNÍ)

PRAVÁ: $X \rightarrow wY$ $X, Y \in N$

NEBO $w \in T^*$

$X \rightarrow w$

GRAMATIKA TYPU 3 (LINEÁRNÍ)

LEVÁ: $X \rightarrow Yw$ $X, Y \in N$

NEBO $w \in T^*$

$X \rightarrow w$

POZNÁMKA: POROVNÁNÍ KONTEXTOVÝCH A BEZKONTEXTOVÝCH GRAMATIK

KONTEXTOVÁ: $\alpha X \beta \rightarrow \alpha \gamma \beta$

BEZKONTEXTOVÁ: $X \rightarrow \gamma$

V KG LZE X PŘEPSAT NA γ JEN V KONTEXTU $\alpha \beta$.

U BKG LZE X PŘEPSAT NA γ KDYKOLI.

GRAMATIKA JE TYPU i, JESTLIŽE JSOU VŠECHNA PŘEPISOVACÍ PRAVIDLA TYPU i NEBO VYŠŠÍHO.

JINAK ŘEČENO: O TYPU GRAMATIKY ROZHODUJE „NEJHORŠÍ" PRAVIDLO.

POZOR: „SMÍCHÁNÍM" PRAVIDEL PRAVÝCH LINEÁRNÍCH A LEVÝCH LINEÁRNÍCH VZNIKNE BEZKONTEXTOVÁ GRAMATIKA.

PŘ:
$$\left.\begin{array}{l} S \to \overset{3P}{abA} \mid \overset{3}{ab} \\ A \to \overset{3P}{aaB} \mid \overset{3}{ba} \\ B \to \overset{2}{AB} \mid \overset{3}{e} \end{array}\right\}$$
GRAMATIKA TYPU 2

PŘ:
$$\left.\begin{array}{l} S \to \overset{3P}{abA} \mid \overset{3}{ab} \\ A \to \overset{3L}{Bab} \mid \overset{3}{e} \\ B \to \overset{3P}{bbB} \mid \overset{3}{b} \end{array}\right\}$$
GRAMATIKA TYPU 2

PODOBNĚ JAKO GRAMATIKY SE KLASIFIKUJÍ I JAZYKY.

JAZYK L JE TYPU i, JESTLIŽE EXISTUJE GRAMATIKA G TYPU i TAKOVÁ, ŽE $L = L(G)$.

MÁ SMYSL UVAŽOVAT NEJVYŠŠÍ TYP, TEDY MAXIMÁLNÍ i.

Př: G: $S \rightarrow AB$

$A \rightarrow aaA \mid e$

$B \rightarrow bbb$

G JE TYPU 2.

JAKÉHO TYPU JE $L(G)$?

PROVEDEME ODVOZENÍ NĚKOLIKA ŘETĚZCŮ:

$S \Rightarrow AB \Rightarrow B \Rightarrow bbb$

$S \Rightarrow AB \Rightarrow aaAB \Rightarrow aaB \Rightarrow aabbb$

$S \Rightarrow AB \Rightarrow aaAB \Rightarrow aaaaAB \Rightarrow$

$\Rightarrow aaaaB \Rightarrow aaaabbb$

$S \Rightarrow AB \Rightarrow aaAB \Rightarrow aaaaAB \Rightarrow$

$\Rightarrow aaaaaaAB \Rightarrow aaaaaaB \Rightarrow$

$\Rightarrow aaaaaabbb$

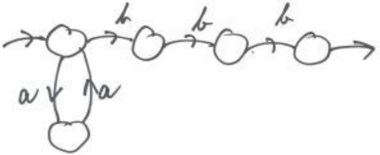
$\vdots$

ZŘEJMĚ

$L(G) = \{bbb, aabbb, aaaabbb, \ldots\}$

$L(G) = \{w \mid w = a^{2n}bbb \wedge n \geq 0\}$

TENTO JAZYK LZE ROZPOZNAT KONEČNÝM AUTOMATEM.



STEJNÝ JAZYK ALE GENERUJE

I GRAMATIKA G': $S \to aaS \mid bbb$

KTERÁ JE TYPU 3.

<u>ZÁVĚR</u>: JAZYK $L(G)$ JE TYPU 3,

PŘESTOŽE GRAMATIKA G JE TYPU 2.

[EXISTUJE GRAMATIKA VYŠŠÍHO TYPU, KTERÁ GENERUJE STEJNÝ JAZYK. GRAMATIKA G JE „ZBYTEČNĚ SLOŽITÁ".]

TYP JAZYKA URČUJE „VÝPOČETNÍ SÍLU" STROJE, KTERÝ JE ZAPOTŘEBÍ K ROZPOZNÁNÍ JAZYKA (TJ. K ŘEŠENÍ PROBLÉMU SYNTAKTICKÉ ANALÝZY).

TYP JAZYKA	SYNTAKTICKÝ ANALYZÁTOR
0	TURINGŮV STROJ
1	LINEÁRNĚ OMEZENÝ AUTOMAT
2	NEDETERMINISTICKÝ ZÁSOBNÍKOVÝ AUTOMAT
3	KONEČNÝ AUTOMAT

FJP

TI

NEJVĚTŠÍ PRAKTICKÉ POUŽITÍ MAJÍ JAZYKY TYPU 2.

- ROZPRACOVANÉ METODY PŘEKLADU
- PRO VHODNĚ NAVRŽENÉ JAZYKY JE SYNTAKTICKÁ ANALÝZA DETERMINISTICKÁ
- MODERNÍ VYŠŠÍ PROGRAMOVACÍ JAZYKY JSOU TYPU 2

JAZYKY TYPU 3

- K POPISU OBJEKTŮ PŘI ROZPOZNÁVÁNÍ SCÉNY, AKUSTICKÝCH SIGNÁLŮ APOD. (JEDNODUCHÉ ÚLOHY UMĚLÉ INTELIGENCE)
- LEXIKÁLNÍ ANALÝZA (ROZPOZNÁVÁNÍ KLÍČOVÝCH SLOV, IDENTIFIKÁTORŮ A KONSTANT V PROGRAMU)

JAZYKY TYPU 3

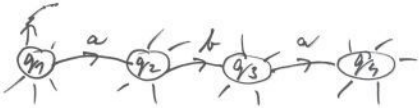
V DALŠÍM UKÁŽEME, ŽE KAŽDÝ JAZYK TYPU 3 JE ROZPOZNATELNÝ KONEČNÝM AUTOMATEM.

NEJDŘÍVE PRECIZUJEME, CO JE TO JAZYK AKCEPTOVANÝ AUTOMATEM (TJ. ROZPOZNÁVANÝ AUTOMATEM).

PŘECHODOVÁ FUNKCE : $\delta : Q \times \Sigma \to Q$

ZOBECNĚNÁ PŘECHODOVÁ FUNKCE : $\delta^* : Q \times \Sigma^* \to Q$

DEFINUJE, JAK AUTOMAT ZAREAGUJE NIKOLI NA VSTUPNÍ PÍSMENO, ALE NA CELÝ VSTUPNÍ ŘETĚZEC.



PROTOŽE

$$\delta(q_1, a) = q_2$$

$$\delta(q_2, b) = q_3$$

$$\delta(q_3, a) = q_4$$

JE

$$\delta^*(q_1, aba) = q_4$$

ZOBECNĚNÁ PŘECHODOVÁ FUNKCE δ^* JE JEDNOZNAČNĚ URČENA PŘECHODOVOU FUNKCÍ δ.

δ^* JE ROZŠÍŘENÍM FUNKCE δ.

DEFINIČNÍ OBOR PŘECHODOVÉ FUNKCE:

$$D(\delta) = Q \times \Sigma$$

DEFINIČNÍ OBOR ZOBECNĚNÉ PŘECH. FUNKCE:

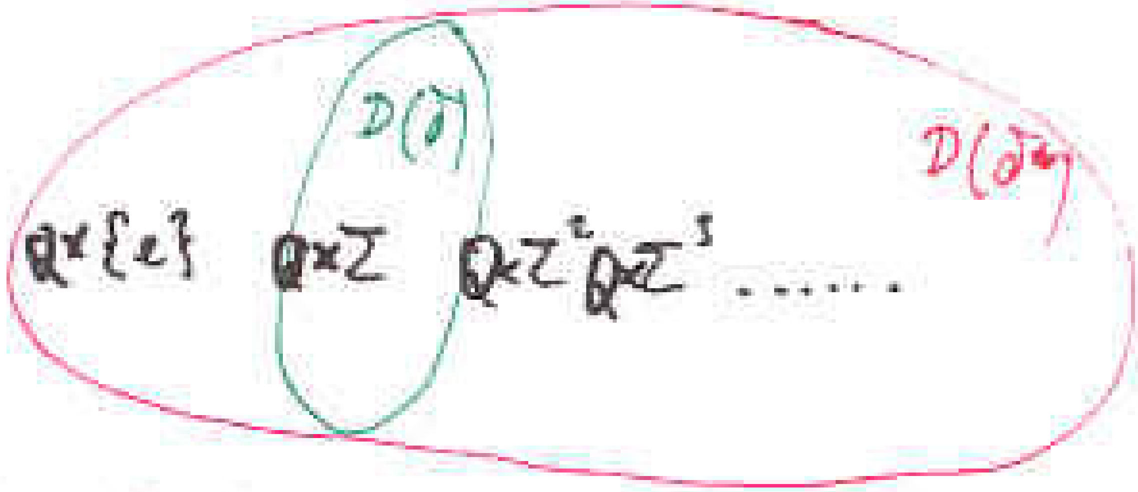
$$D(\delta^*) = Q \times \Sigma^* = Q \times (\{e\} \cup \Sigma \cup \Sigma^2 \cup \ldots) =$$

$$= Q \times \bigcup_{i=0}^{\infty} \Sigma^i$$

ZŘEJMĚ $D(\delta) \subseteq D(\delta^*)$

A MUSÍ PLATIT

$$\delta^*(q,a) = \delta(q,a) \quad \forall q \in Q \quad \forall a \in \Sigma$$



KONEČNÁ

NEKONEČNÁ SPOČETNÁ

δ^* SE DEFINUJE REKURZIVNĚ POMOCÍ δ :

$$\delta^*(q, wa) = \delta(\delta^*(q, w), a)$$

$$\forall q \in Q \quad \forall w \in \Sigma^* \quad \forall a \in \Sigma$$

$$\delta^*(q, e) = q \qquad \forall q \in Q$$

JAZYK ROZPOZNÁVANÝ AUTOMATEM

JAZYKEM ROZPOZNÁVANÝM KONEČNÝM AUTOMATEM $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$ NAZÝVÁME JAZYK

$$L(\mathcal{A}) = \{ w \mid w \in \Sigma^* \land \delta^*(q_0, w) \in F \}$$

EKVIVALENTNÍ AUTOMATY

AUTOMATY

$$A_1 = (Q_1, \Sigma, \delta_1, q_{01}, F_1)$$

$$A_2 = (Q_2, \Sigma, \delta_2, q_{02}, F_2)$$

JSOU EKVIVALENTNÍ PRÁVĚ TEHDY, KDYŽ

$$L(A_1) = L(A_2)$$

POZOR – EKVIVALENTNÍ MOHOU BÝT I TAKOVÉ AUTOMATY, KTERÉ NEMAJÍ STEJNÝ POČET STAVŮ.

REDUKOVANÝ AUTOMAT – TAKOVÝ REPREZENTANT TŘÍDY EKVIVALENTNÍCH AUTOMATŮ, KTERÝ MÁ MINIMÁLNÍ POČET STAVŮ (EXISTUJÍ ALGORITMY MINIMALIZACE).

ŘETĚZEC w JE **PŘIJÍMÁN** (**AKCEPTOVÁN**) AUTOMATEM $\mathcal{A}$ PRÁVĚ TEHDY, KDYŽ $w \in L(\mathcal{A})$. V OPAČNÉM PŘÍPADĚ JE ŘETĚZEC w ZAMÍTNUT.

V DALŠÍM SE BUDEME SNAŽIT UKÁZAT SOUVISLOST GRAMATIK A JAZYKŮ TYPU 3 A KONEČNÝCH AUTOMATŮ.

MOTIVAČNÍ PŘÍKLAD:

$$
\begin{aligned}
G: \quad & S \to \overset{1}{0A} \mid \overset{2}{1S} \mid \overset{3}{e} \\
& A \to \overset{4}{0B} \mid \overset{5}{1A} \\
& B \to \overset{6}{0S} \mid \overset{7}{1B}
\end{aligned}
$$

G JE ZŘEJMĚ GRAMATIKA TYPU 3 (PRAVÁ LINEÁRNÍ).

ZKONSTRUUJEME „AUTOMAT" A TÍMTO FORMÁLNÍM POSTUPEM:

- STAVY BUDOU ODPOVÍDAT NETERMINÁLNÍM SYMBOLŮM
- VSTUPY BUDOU ODPOVÍDAT TERMINÁLNÍM SYMBOLŮM
- PŘECHODOVOU FUNKCI ZKONSTRUUJEME NA ZÁKLADĚ ANALOGII

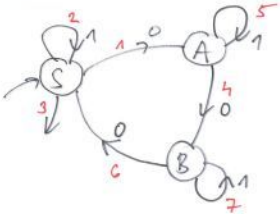
$$X \to aY \in P \implies$$

$$X \xrightarrow{a} Y$$ BUDE PŘECHOD

- POČÁTEČNÍ STAV BUDE ODPOVÍDAT POČÁTEČNÍMU SYMBOLU
- MNOŽINU KONCOVÝCH STAVŮ URČÍME TAK, ŽE

$$X \to e \in P \implies X \in F$$

VÝSLEDEK:



TENTO GRAF JE (SHODOU OKOLNOSTÍ, NIKOLI OBECNĚ) PŘECHODOVÝM GRAFEM KONEČNÉHO AUTOMATU.

BUDEME ZKOUMAT GENEROVÁNÍ ŘETĚZCŮ GRAMATIKOU A ZPRACOVÁNÍ TĚCHTO ŘETĚZCŮ AUTOMATEM.