

Cvičení k přednášce NTIN071 Automaty a gramatiky

Marta Vomlelová

17. dubna 2026

Seznam cvičení

1	Konečné automaty - úvod	2
2	Redukt DFA, Iterační lemma	3
3	Nedeterministické konečné automaty, Myhill-Nerode, PL	4
4	Regulární výrazy	5
5	Uzávěrové vlastnosti, 2DFA (+Mealy, Moore stroj)	7
6	Gramatiky úvod	9
7	Iterační lemma pro CFL, (+ Redukce, ChNF gramatiky)	10
8	Zásobníkové automaty (PDA), Deterministické PDA	12
9	Uzávěrové vlastnosti, převody PDA - CFG, Dvousměrné DFA	13
10	Turingovy stroje	14
11	Úvod do složitosti	15

Podmínky zápočtu

Zápočet se uděluje za dosažení dostatečného počtu bodů z aktivity v semestru (max 20 bodů) a testu na konci semestru (max 120 bodů). Na zápočet je potřeba 80 bodů. Test lze nejvýše dvakrát opakovat (jeden opravný termín vypsan cvičícím, případný další už sběrný přes všechny paralelky).

Seznam možných úkolů v zápočtové písemce

Jazyk, automat či gramatika může být libovolná, popsána slovně, automatem, gramatikou, regulárním výrazem, pomocí operací sjednocení, doplňku, průniku, reverze, zřetězení apod.

1. Trasujte výpočet daného automatu (DFA, NFA, PDA, TM) nad daným slovem, najděte derivaci daného slova v dané gramatice,
2. Rozhodněte, zda je jazyk L regulární a svou odpověď dokažte (buď použitím iteračního lemmatu, Myhill-Nerodovy věty, uzávěrových vlastností nebo sestrojením konečného automatu či regulární gramatiky).
3. Rozhodněte, zda je jazyk L bezkontextový a svou odpověď dokažte (buď použitím iteračního lemmatu, uzávěrových vlastností nebo sestrojením zásobníkového automatu či bezkontextové gramatiky).
4. Navhňte konečný automat přijímající jazyk L .
5. Navhňte zásobníkový automat přijímající jazyk L .
6. Převedte ϵ nedeterministický konečný automat na deterministický přijímající stejný jazyk.
7. Najděte redukt konečného automatu, najděte ekvivalentní stavy konečného automatu.
8. Napište gramatiku generující jazyk L .
9. Popište Turingův stroj přijímající konkrétní jazyk.

10. Jakýkoli jazyk může být definován slovně, množinově, množinovými a řetězcovými operacemi, automatem (DFA, NFA, PDA, TM), regulárním výrazem či gramatikou, tedy můžete dostat úkol převodu mezi jednotlivými reprezentacemi.

Pro informaci:

1. Mealy a Moorův stoj, CYK algoritmus nebude v zápočtové písemce ani u zkoušky.

1 Konečné automaty - úvod

1. Navrhněte konečné automaty na počítání symbolů ve slovech. Pro popis automatů použijte ohodnocený graf (stavový diagram) nebo tabulku.

- (a) $L = \{w \mid w \in \{a, b\}^* \ \& \ (\exists k \in \mathbb{N}_0) |w|_a = 3k\}$
- (b) $L = \{w \mid w \in \{a, b\}^* \ \& \ (\exists k \in \mathbb{N}_0) |w|_a = 2k\}$
- (c) $L = \{w \mid w \in \{a, b\}^* \ \& \ [(\exists k \in \mathbb{N}_0) |w|_a = 3k \vee (\exists \ell \in \mathbb{N}_0) |w|_a = 2\ell]\}$
- (d) $L = \{w \mid w \in \{a, b\}^* \ \& \ (\exists k \in \mathbb{N}_0) |w|_a = 3k \ \& \ (\exists \ell \in \mathbb{N}_0) |w|_a = 2\ell\}$
- (e) $L = \{w \mid w \in \{a, b\}^* \ \& \ (\exists k \in \mathbb{N}_0) |w|_a \neq 3k \ \& \ (\forall \ell \in \mathbb{N}_0) |w|_b = 2\ell\}$

2. Navrhněte konečné automaty, které přijímají slova obsahující určité podslovo:

- (a) $L = \{w \mid w \in \{a, b\}^* \ \& \ (\exists u \in \{a, b\}^*) w = abba.u\}$ Jazyk obsahuje slova, která začínají *abba*.
- (b) $L = \{w \mid w \in \{a, b\}^* \ \& \ (\exists u \in \{a, b\}^*) w = u.abba\}$ Jazyk obsahuje slova, která končí *abba*.
- (c) $L = \{w \mid w \in \{a, b\}^* \ \& \ (\exists u, v \in \{a, b\}^*) w = u.abba.v\}$ Jazyk obsahuje slova, v nichž se vyskytuje podslovo *abba*.
- (d) $L = \{w \mid w \in \{a, b\}^* \ \& \ (\exists u \in \{a, b\}^*) w = u.ab \ \& \ (\exists k \in \mathbb{N}_0) |w| = 3k + 1\}$
- (e) $L = \{w \mid w \in \{a, b\}^* \ \& \ [(\exists u \in \{a, b\}^*) w = u.ab \ \vee \ (\exists k \in \mathbb{N}_0) |w| = 3k + 1]\}$.

3. Navrhněte konečné automaty, které přijímají slova, u nichž začátek souvisí s koncem:

- (a) $L = \{w \mid w \in \{a, b\}^* \ \& \ (\exists u, v \in \{a, b\}^*) (|u| = 1 \ \& \ w = u.v.u)\}$. Jazyk obsahuje slova, která mají délku aspoň 2 a začínají a končí stejným symbolem.
- (b) $L = \{w \mid w \in \{a, b\}^* \ \& \ (\exists u, v \in \{a, b\}^*) (|u| = 2 \ \& \ w = u.v.u)\}$. Jazyk obsahuje slova, která mají délku aspoň 4 a začínají a končí stejnou (uspořádanou) dvojicí symbolů.
- (c) $L = \{w \mid w \in \{a, b\}^* \ \& \ (\exists u, v, z \in \{a, b\}^*) (|u| = 2 \ \& \ w = u.v \ \& \ w = z.u)\}$. Jazyk obsahuje slova, která začínají a končí stejnou (uspořádanou) dvojicí symbolů.
- (d) $L = \{w \mid w \in \{a, b\}^* \ \& \ (\exists u, v, z \in \{a, b\}^*) (|u| = 2 \ \& \ |z| = 2 \ \& \ w = u.v.z \ \& \ u \neq z)\}$. Jazyk obsahuje slova, která mají délku aspoň 4 a začínají a končí různými (uspořádanými) dvojicemi symbolů.

4. Vyjádřete množinově (jako v předchozích úlohách) jazyky, které jsou přijímány následujícími automaty popsanými tabulkou:

(a)

	0	1
$\rightarrow^* p$	q	p
q	r	q
r	p	r

(b)

	0	1
$\rightarrow p$	q	p
* q	r	q
* r	p	r

(c)

	0	1
$\rightarrow p$	p	q
q	p	r
* r	p	r

(d)

	0	1
$\rightarrow p$	p	q
* q	r	q
* r	p	q

5. Necht posloupnost kóduje průběh tenisového zápasu. Z hlediska prvního hráče necht 1 znamená získání bodu, necht 0 znamená získání bodu protihráčem. Navrhněte konečný automat, který přijímá posloupnost 0,1 kódující hru, právě když kód odpovídá výhře prvního hráče. (stačí jeden game).

2 Redukt DFA, Iterační lemma

1. Z následujících automatů odstraňte nedosažitelné stavy. Ve výsledných automatech najděte množiny ekvivalentních stavů. Zkonstruujte redukty automatů.

(a)

	a	b
$\rightarrow^* 0$	0	5
1	1	3
2	2	7
3	3	2
* 4	4	1
5	5	1
* 6	6	2
7	7	0

(b)

	a	b
A	A	F
B	B	A
C	C	D
D	D	B
E	E	C
$\rightarrow^* F$	F	E

(c)

	a	b
$\rightarrow 1$	2	3
2	2	4
* 3	3	5
4	2	7
* 5	6	3
* 6	6	6
7	7	4
8	2	3
9	9	4

(d)

	a	b
* A	H	G
B	B	A
C	E	D
D	D	B
E	C	D
F	F	E
G	G	F
$\rightarrow^* H$	A	G

(e)

	a	b
$\rightarrow^* 0$	1	2
1	3	0
2	4	5
3	0	2
4	2	5
5	0	3

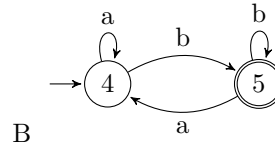
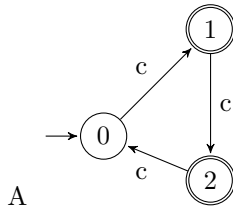
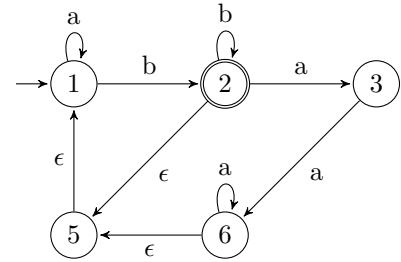
(f)

	a	b
$\rightarrow 0$	1	2
1	0	3
2	4	1
3	0	1
* 4	2	2
5	4	3

- Jaké je nejkratší slovo (pokud existuje), které rozlišuje stavy
 - 1 a 5 u automatu a)?
 - 2 a 4 u automatu c)?
 - 3 a 5 u automatu e)?
- Rozhodněte, zda jsou některé z automatů a)-d) z předchozího cíčení po dvojicích ekvivalentní. Pokud ne, najděte slovo patřící do jazyka právě jednoho automatu.
- Definujte automatový homomorfismus. Najděte homomorfismus automatu
- Automaty jsou izomorfní, pokud existuje homomorfismus který je zároveň bijekcí stavů. Navrhněte dvojici konečných automatů, které jsou
 - redukované a neizomorfní
 - ekvivalentní a neizomorfní.
- Formulujte iterační lemma pro regulární jazyky.
- Formulujte Myhill-Nerodovu větu.
- Rozhodněte, zda jsou následující jazyky regulární:
 - $L = \{ww \mid w \in \{a, b\}^*\}$
 - $L = \{ww \mid w \in \{a\}^*\}$
 - $L = \{a^i b^j \mid i, j = 0, 1, 2, \dots\}$
 - $L = \{a^i a^i b^j \mid i, j = 0, 1, 2, \dots\}$
 - $L = \{a^i b^j a^i \mid i, j = 0, 1, 2, \dots\}$
 - $L = \{a^i b^i a^j \mid i, j = 0, 1, 2, \dots\}$
 - $L = \{a^i b^j c^k \mid i, j, k = 0, 1, 2, \dots\}$
 - $L = \{ww^R \mid w \in \{a, b\}^*\}$
 - $L = \{ww \mid w \in \{a, b\}^* \text{ \& } |w|_a = |w|_b\}$
 - $L = \{a^{2i} \mid i = 0, 1, 2, \dots\}$
 - $L = \{a^{i^2} \mid i = 0, 1, 2, \dots\}$
 - $L = \{a^p \mid p \text{ je prvočíslo}\}.$

3 Nedeterministické konečné automaty, Myhill-Nerode, PL

1.
 - (a) K nedeterministickému automatu s přechodovou funkcí na obrázku sestrojte ekvivalentní deterministický konečný automat.
 - (b) Deterministický FA zredukujte.
 - (c) Je výsledkem podmnožinové konstrukce vždy redukovaný automat?
2. Mějme automaty A, B na obrázku.



Zkonstruuje nedeterministický konečný automat, který přijímá jazyk:

- (a) $L(A).L(B)$
 - (b) $L(A)^*$
 - (c) $L(A)^R$
 - (d) $\overline{L(A).L(B)}$.
- Sestrojte příslušné deterministické automaty. Automat (2b) zredukujte.
3. Určete všechny třídy ekvivalence z Nerodovy věty pro následující jazyky:
 - (a) $L_1 = \{a, aab, abb\}$
 - (b) $L_2 = \{a^m ba^n | m, n \in \mathbb{N}_1\}$
 - (c) $L_3 = \{a^n ba^n | n \in \mathbb{N}_0\}$.
4. Pokud jste ve druhém cvičení nedokončili příklad 8, vraťte se k němu.
5. Je jazyk $L_c = \{(ab)^n a (ba)^n | n \in \mathbb{N}_1\}$ regulární? Dokažte.
6. Je jazyk $L_d = \{(ba)^n a (ba)^n | n \in \mathbb{N}_1\}$ regulární? Dokažte.
7. (Náročnější) Uvažujme jazyk

$$L = \{ab(ba)^m ba^n ba^n | m, n \in \mathbb{N}_1\} \cup \{abba^m ba^n | m, n \in \mathbb{N}_1\}$$

- (a) Dokažte, že splňuje iterační lemma a najděte konstantu n .
 - (b) Dokažte, že není regulární pomocí Myhill-Nerodovy věty.
8. Nechť A a B jsou konečné automaty pracující nad abecedou Σ . Navrhněte algoritmus, který rozhodne zda:
 - (a) $L(A) = \emptyset$
 - (b) $L(A) = L(B)$
 - (c) $L(A) \subset L(B)$
 - (d) $L(A) = \Sigma^*$
 - (e) $L(A)$ je nekonečný.

4 Regulární výrazy

1. Navrhněte regulární výrazy reprezentující následující jazyky nad $\Sigma = \{a, b\}$:

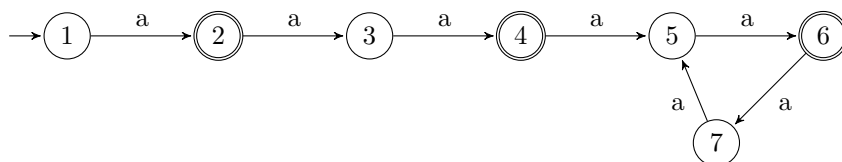
- (a) jazyk sestávající ze slov, která obsahují *abba* jako podslovo
- (b) jazyk sestávající ze slov, která mají prefix *abb* a sufix *bbaa*
- (c) jazyk sestávající ze slov, kde počet výskytů *a* je dělitelný 3
- (d) jazyk sestávající ze slov, která začínají a končí stejnou dvojicí symbolů
- (e) jazyk sestávající ze slov, která neobsahují podslovo *aa*

2. Pro následující regulární výrazy zkonstruujte konečné automaty, které přijímají jimi reprezentované jazyky:

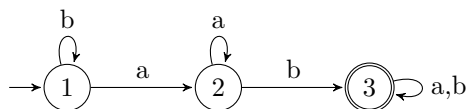
- (a) $\mathbf{ab + ba}$
- (b) $\mathbf{x\ a^2 + b^2 + ab}$
- (c) $\mathbf{x\ a + b^*}$
- (d) $\mathbf{x\ (ab + c)^*}$
- (e) $\mathbf{x\ ((ab + c)^+ a(bc)^* + b)^*}$
- (f) $\mathbf{((ab + c)^* a(bc)^* + b)^*}$
- (g) $\mathbf{x\ (01^* + 101)^* 0^* 1}$
- (h) $\mathbf{x\ (01)^* 11(01)^* (0 + 1)^* 00}$

3. Pro následující konečné automaty sestojte regulární výrazy, které reprezentují stejný jazyk:

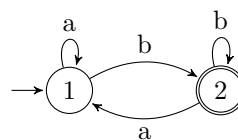
(a)



(b)

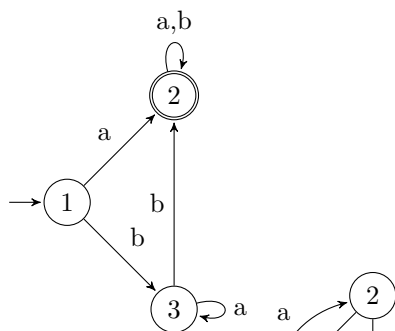


(e)

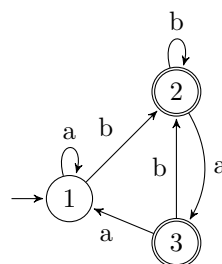
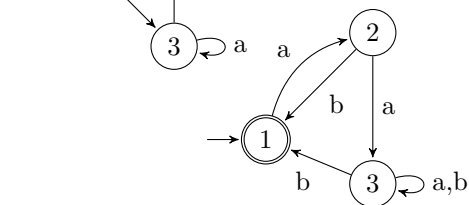


(f)

(c)



(d)



(g) Navrhněte algoritmus pro rozhodování, zda je dvojice vstupních regulárních výrazů ekvivalentní, tj. zda reprezentují stejný jazyk. Algoritmus aplikujte na dvojici regulárních výrazů:

$$(a + b)(a + b)^* a \text{ a } a(a + b)^* + b(a + b)^*$$

4. Napište regulárním výrazem možné formáty telefonního čísla, včetně mezinárodního směrování. Představme si navíc, že země s volačkou '11' má desetimístné národní číslo, země s volačkou '88' jen sedmimístné (případně doplňte podle skutečnosti).
5. Zapište formát přípustného identifikátoru proměnné. Můžete používat intervaly a-z apod.
6. Zapište možné formáty velikosti paměti počítače, možné jednotky Mb a Gb, číslo může být na jedno desetinné místo.
7. Uvažujme zjednodušené HTML, kde jsou k dispozici konstrukce

```
<p>      </p>
<a>      </a>
<table>  </table>
<tr>     </tr>
<td>     </td>
```

všechny bez atributů. Využijte tvrzení o regulární substituci k rozhodnutí, zda texty ve formátu zjednodušeného HTML tvoří regulární jazyk.

5 Uzávěrové vlastnosti, 2DFA (+Mealy, Moore stroj)

- Nechť h je homomorfismus $h : \{0, 1, 2\} \rightarrow \{a, b\}$, $h(0) = a$, $h(1) = ab$, $h(2) = ba$.
 - Určete $h(0120)$.
 - Určete $h(21120)$.
 - Mějme $L = L(\mathbf{01^*2})$. Určete $h(L)$.
 - Mějme $L = L(\mathbf{0 + 12})$. Určete $h(L)$.
- Mějme jazyk L nad abecedou $\Sigma = a, b$. Levá derivace jazyka slovem $v \in \Sigma^*$ je definovaná $\partial_v(L) = \{w \mid vw \in L\}$, pravá derivace $\partial_v^R(L) = \{w \mid wv \in L\}$. Dokažte, že je-li L regulární, je regulární i jeho levá i pravá derivace.
- Mějme jazyky L, M nad abecedou $\Sigma = a, b$.
Quocient jazyka $M \setminus L = \{w \mid (\exists v \in M)vw \in L\}$, $L/M = \{w \mid (\exists v \in M)wv \in L\}$. Dokažte, že je-li L regulární, je regulární i jeho levý i pravý quotient.
- Dokažte, že regulární jazyky jsou uzavřené na operace:
 - $\min(L) = \{w \mid w \in L \ \& \ \neg(\exists v \in L, u \in \Sigma^+ : w = vu)\}$, tj. žádný vlastní prefix w není z L .
 - $\max(L) = \{w \mid w \in L \ \& \ \neg(\exists v \in L, u \in \Sigma^+ : vu \in L)\}$, tj. w není prefixem jiného slova z L .
 - $\text{init}(L) = \{w \mid u \in \Sigma^* : wu \in L\}$, tj. w je prefixem nějakého slova z L .

Dvousměrné konečné automaty - doplňující, ale nejzajímavější

- Simulujte výpočet dvousměrného automatu na vstupu $\#aaab\#$.

	a	b	#
$\rightarrow p_0$	-	-	p,+1
p	p,+1	p,+1	q,-1
q	qa,-1	qb,-1	-
qa	-	qa,-1	q _F ,+1
qb	qb,-1	-	q _F ,+1
q_F	q _F ,+1	q _F ,+1	q _{FF} ,+1
$*q_{FF}$	-	-	-

- Nechť L je regulární jazyk. Rozhodněte a zdůvodněte, zda je jazyk $K = \{w \mid \#w\$ \in L\}$ regulární.
- Uvažujme konečný automat $A = (Q, \Sigma, \delta, q_0, F)$. Navrhněte dvousměrné automaty (pro L_1 a L_4 nedeterministické), které přijímají jazyky:
 - $L_1 = \{\#w\$ \mid ww^R \in L(A)\}$
 - $L_2 = \{\#w\$ \mid ww \in L(A)\}$
 - $L_3 = \{\#w\$ \mid (\exists v \in \Sigma^*)wv \in L(A) \ \& \ |w| = |v|\}$
 - $L_4 = \{\#w\$ \mid (\exists u, v \in \Sigma^*)w = uv \ \& \ uv^R \in L(A)\}$
- Navrhněte nedeterministický konečný automat přijímající jazyk $L_1 = \{\#w\$ \mid ww^R \in L(A)\}$. Přitom nevyužívejte znalosti dvousměrných automatů.
- Popište princip konstrukce ekvivalentního konečného automatu k danému dvousměrnému automatu.
- Pro jazyky L_1, L_2, L_3, L_4 navrhněte (nedeterministické) konečné automaty, které je přijímají. Při návrhu využijte znalosti dvousměrných automatů.

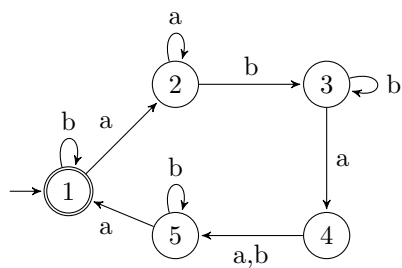
Mealy, Moore stroj - jen doplňující

- Navrhněte převod mezi Mealyho a Mooreovým strojem.
- Navrhněte Mealyho nebo Mooreův stroj, který realizuje operaci:
 - bitového součtu bitových vektorů

(b) bitového součinu bitových vektorů

(c) aritmetického součtu bitových vektorů, když čísla čteme odzadu.

3. Uvažujme konečný automat A zadaný následujícím stavovým diagramem:



Sestrojte konečné automaty, které přijímají jazyky:

- $L_1 = \{w \mid (\exists u, v \in \{a, b\}^*) w = uv \ \& \ uav \in L(A)\}$
- $L_2 = \{w \mid (\exists u, v \in \{a, b\}^*) w = uv \ \& \ (uav \in L(A) \vee ubv \in L(A))\}$
- $L_3 = \{w \mid (\exists u, v \in \{a, b\}^*) w = uav \ \& \ uv \in L(A)\}$

6 Gramatiky úvod

1. Navrhněte gramatiky, které generují následující jazyky. Lze gramatiku zkonstruovat ve všech případech? (Obecnou? Bezkontextovou? Regulární?)

(a) $L = \{a^i b^i \mid i = 0, 1, 2, \dots\}$

(g) $L = \{a^i b^i c^i \mid i = 0, 1, 2, \dots\}$

(b) $L = \{a^i b^j \mid i, j = 0, 1, 2, \dots\}$

(h) $L = \{a^{i^2} \mid i = 0, 1, 2, \dots\}$

(c) $L = \{a^i a^i b^j \mid i, j = 0, 1, 2, \dots\}$

(i) $L = \{ww^R \mid w \in \{a, b\}^*\}$

(d) $L = \{a^i b^j a^i \mid i, j = 0, 1, 2, \dots\}$

(j) $L = \{ww \mid w \in \{a, b\}^*\}$

(e) $L = \{a^i b^i a^j \mid i, j = 0, 1, 2, \dots\}$

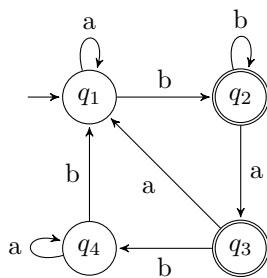
(k) $L = \{ww \mid w \in \{a\}^*\}$

(f) $L = \{a^i b^j a^k \mid i, j, k = 0, 1, 2, \dots\}$

(l) $L = \{a^i b^j c^k \mid i, j, k = 0, 1, 2, \dots \& i \leq j \leq k\}$

- (m) správné uzávorkování, tj. stejně levých a pravých, nikdy v průběhu ne více pravých než levých; '()()' patří do jazyka.

2. Pro následující konečný automat naleznete ekvivalentní gramatiku. V jaké třídě Chomského hierarchie se budete pohybovat?



3. Pro následující gramatiku naleznete ekvivalentní konečný automat. Lze to provést s libovolnou gramatikou?

$$S \rightarrow abS \mid babA \mid \epsilon$$

$$A \rightarrow abA \mid aB \mid bC$$

$$B \rightarrow abS \mid B \mid bC \mid \epsilon$$

$$C \rightarrow aab \mid A \mid aA \mid \epsilon.$$

4. Jaký jazyk generuje gramatika G ? Je gramatika kontextová? Naleznete ekvivalentní kontextovou gramatiku.

$$G = (\{S, A, B, C\}, \{a, b, c\}, P, S), \text{ kde } P = \left\{ \begin{array}{l} S \rightarrow aSBC \mid \epsilon \\ B \rightarrow BBC \\ C \rightarrow CC \\ CB \rightarrow BC \\ aB \rightarrow ab \\ bB \rightarrow bb \\ bC \rightarrow bc \\ cC \rightarrow cc \end{array} \right\}.$$

7 Iterační lemma pro CFL, (+ Redukce, ChNF gramatiky)

1. Rozhodněte, zda jsou následující jazyky bezkontextové (případně zopakujte, zda jsou regulární).

- (a) $L = \{ww \mid w \in \{a, b\}^*\}$
- (b) $L = \{a^i b^i \mid i = 0, 1, 2, \dots\}$
- (c) $L = \{a^i b^j a^i \mid i, j = 0, 1, 2, \dots\}$
- (d) $L = \{a^i b^j a^k \mid i, j, k = 0, 1, 2, \dots\}$
- (e) $L = \{a^i b^i c^i \mid i = 0, 1, 2, \dots\}$
- (f) $L = \{a^i b^j c^k \mid i, j, k = 0, 1, 2, \dots \& i \leq j \leq k\}$
- (g) $L = \{ww^R \mid w \in \{a, b\}^*\}$
- (h) $L = \{ww^R \mid w \in \{a, b\}^* \& |w|_a = |w|_b\}$
- (i) $L = \{a^{i^2} \mid i = 0, 1, 2, \dots\}$
- (j) $L = \{a^{i^2+i+1} \mid i = 0, 1, 2, \dots\}$
- (k) $L = \{a^p \mid p \text{ je prvočíslo}\}$
- (l) $L = \{w \mid w \in \{0, 1\}^* \& |w|_0 \neq |w|_1\}$
- (m) $L = \{0^i 1^i 0^j \mid i, j \in \mathbb{N}_0\} \cup \{0^i 1^j 0^j \mid i, j \in \mathbb{N}_0\}$
- (n) $L = \{0^i 1^i 0^j \mid i, j \in \mathbb{N}_0\} \cap \{0^i 1^j 0^j \mid i, j \in \mathbb{N}_0\}$.

Redukce CFG gramatiky, Uzávěrové vlastnosti

2. Zredukujte následující gramatik, tj. odstraňte zbytečné neterminály a je obsahující pravidla.

$$G_1 = (\{S, A, B, C, D, E\}, \{a, b, c\}, P, S), \text{ kde } P = \left\{ \begin{array}{l} S \rightarrow aA|bB|aSa|bSb|\epsilon \\ A \rightarrow bCD|Dba \\ B \rightarrow bB|AC \\ C \rightarrow aA|AC \\ D \rightarrow DE \\ E \rightarrow \epsilon \end{array} \right\}.$$

3. Následující bezkontextovou gramatiku převeďte na redukovanou gramatiku bez ϵ pravidel a jednotkových pravidel, která generuje stejný jazyk.

$$(a) \ G_2 = (\{S, A, B, C\}, \{a, c\}, P, S), \text{ kde } P = \left\{ \begin{array}{l} S \rightarrow aACa \\ A \rightarrow a|B \\ B \rightarrow C|c \\ C \rightarrow cC|\epsilon \end{array} \right\},$$

$$(b) \ G_3 = (\{S, A, C\}, \{0, 1, 2\}, P, S), \text{ kde } P = \left\{ \begin{array}{l} S \rightarrow ASC|\epsilon \\ A \rightarrow 1A0|\epsilon \\ C \rightarrow 2C|S \end{array} \right\},$$

4. Následující bezkontextovou gramatiku převeďte do Chomského normální formy.

$$G_4 = (\{S, E, F\}, \{(\cdot), *, +, a\}, P, S), \text{ kde } P = \left\{ \begin{array}{l} S \rightarrow (E) \\ E \rightarrow F + F|F * F \\ F \rightarrow S|a \end{array} \right\}.$$

5. Navrhněte algoritmus, jak pro dané slovo $w \in \Sigma$ a gramatiku $G = (V, T, P, S)$ rozhodnout, zda slovo w patří do jazyka $L(G)$. (Algoritmus nemusí být efektivní, může mít exponenciální složitost. CYK algoritmus s polynomiální složitostí není povinný, ukážeme pouze, pokud zbyde čas.
6. Mějme abecedu $\Sigma = \{a, b\}$. Je jazyk korektních regulárních výrazů nad Σ regulární? Bezkontextový? Existuje pro něj jednoznačná gramatika?
7. Uvažujme jazyk $L_{11} = \{0, 1\}^* . 11 . \{0, 1\}^*$.

- (a) Je jazyk regulární? Dokažte.
- (b) Je jazyk $\overline{L_{11}}$ regulární? Dokažte.

8. (*) Uvažujeme jazyk přiřazení výrazu do proměnné, $L_p = L(I) \cdot L(E)$, kde $L(I)$ je jazyk možných identifikátorů a $L(E)$ jazyk možných aritmetických výrazů.
- (a) Naznačte gramatiku generující jazyk L_p .
 - (b) Je jazyk regulární? Dokažte.
 - (c) Je jazyk bezkontextový? Dokažte.
 - (d) Je určete levý kvocient $L(I) \setminus L_p$. Je jazyk regulární, bezkontextový? Navrhněte zásobníkový automat, který ho přijímá.
 - (e) Jak bychom pomocí kvocientu vyjádřili jazyk $L(E)$?

8 Zásobníkové automaty (PDA), Deterministické PDA

1. Navrhněte zásobníkové automaty pro následující jazyky (nebo zdůvodněte, proč neexistuje):

- (a) $L_1 = \{w2w^R \mid w \in \{0,1\}^*\}$
- (b) $L_2 = \{ww^R \mid w \in \{0,1\}^*\}$
- (c) $L_3 = \{w \mid w \in \{0,1\}^* \& |w|_0 = |w|_1\}$
- (d) $L_4 = \{u2v \mid u, v \in \{0,1\}^* \& |u| \neq |v|\}$
- (e) $L_i = \{u2v \mid u, v \in \{0,1\}^* \& u[i] \neq v[i]\}$
- (f) $L_5 = \{u2v^R \mid u, v \in \{0,1\}^* \& u \neq v\}$
- (g) $L_6 = \{a^i b^j c^{i+j} \mid i, j = 0, 1, 2, \dots\}$
- (h) $L_7 = \{a^i b^j c^{i*j} \mid i, j = 0, 1, 2, \dots\}$.

2. Definujme operaci slévání $\sqcup$ jako slévání dvou pruhů na dálnici - ne nutně jedno a jedno auto, libovolné množství aut z pruhu za sebou, tj.

$$aab \sqcup cc = \{ccaab, cacab, caacb, caabc, accab, acacb, acabc, aaccb, aacbc, aabcc\}.$$

Pořadí v rámci pruhů zůstává zachováno. Pro jazyky definujeme

$$L_1 \sqcup L_2 = \bigcup_{u \in L_1, v \in L_2} (u \sqcup v).$$

- (a) Určete $L_1 \sqcup L_2$ pro $L_1 = \{a^i b^i \mid i \in \mathbb{N}\}$ a $L_2 = \{c^k \mid k \in \mathbb{N}\}$.
 - (b) Dokažte, že je jazyk $L_1 \sqcup L_2$ bezkontextový.
 - (c) Je jazyk $L_1 \sqcup L_2$ bezkontextový pro L_1 regulární a L_2 bezkontextový?
3. Najděte podmínky, podle kterých lze určit, zda je jazyk generovaný danou bezkontextovou gramatikou neko-
nečný. Inspirujte se pumping lemmatem pro bezkontextové jazyky.

9 Uzávěrové vlastnosti, převody PDA - CFG, Dvousměrné DFA

- Jsou jazyky bezkontextové? Odpověď zdůvodněte.
 - $L_n = \{a^i b^j c^k \mid i \neq k \vee i \neq j \vee j \neq k\}$.
 - $\overline{L_3} = \{a^i b^j c^i \mid i \in \mathbb{N}\}$ (stačí naznačit).
 - $\overline{L_{\leq}} = \{a^i b^j c^k \mid 0 \leq i \leq j \leq k\}$ (stačí naznačit).
 - $L = \{0^i 1^j 2^k 3^l \mid i = 0 \vee j = k = l\}$.
 - $L = \{ww^R \mid w \in \{a, b\}^* \& |w|_a = |w|_b\}$.
 - $L = \{ww^R \mid w \in \{a, b\}^* \vee |w|_a = |w|_b\}$.
- Uvažujeme jazyk $L_{01} = \{0^i 1^i \mid i \in \mathbb{N}_0\}$
 - Je jazyk bezkontextový?
 - Je jazyk přijímaný nějakým deterministickým zásobníkovým automatem?
 - Je doplněk tohoto jazyka přijímaný nějakým deterministickým zásobníkovým automatem?
 - Je doplněk tohoto jazyka přijímaný nějakým deterministickým zásobníkovým automatem prázdným zásobníkem?
- Uvažujme $G_5 = (\{E, T, F\}, \{(\cdot), *, +, , 1\}, E, P)$, kde $P = \left\{ \begin{array}{l} E \rightarrow E + T \mid T \\ T \rightarrow T * F \mid F \\ F \rightarrow 1 \mid (E) \end{array} \right\}$. Sestrojte zásobníkové automaty Z_1, Z_2 že $L(Z_1) = L(G)$ a $N(Z_2) = L(G)$.
- Zásobníkový automat pro jazyk $L_3 = \{w \mid w \in \{0, 1\}^* \& |w|_0 = |w|_1\}$ převedte na bezkontextovou gramatiku.
- × Uvažujeme jazyk $L_{w2wr} = \{w2w^R \mid w \in \{0, 1\}^*\}$ nad abecedou $\{0, 1, 2\}^*$.
 - Je jazyk bezkontextový?
 - Je jazyk přijímaný nějakým deterministickým zásobníkovým automatem?
 - Je doplněk tohoto jazyka přijímaný nějakým deterministickým zásobníkovým automatem?
 - Je jazyk $\bar{L} \cap \{0, 1\}^* . 2 . \{0, 1\}^*$ přijímaný nějakým deterministickým zásobníkovým automatem?

Dvousměrné konečné automaty - doplňující, ale nejzajímavější

- Simulujte výpočet dvousměrného automatu na vstupu $\#aaab\#$.

	a	b	#
$\rightarrow p_0$	-	-	p,+1
p	p,+1	p,+1	q,-1
q	qa,-1	qb,-1	-
qa	-	qa,-1	qF,+1
qb	qb,-1	-	qF,+1
qF	qF,+1	qF,+1	qFF,+1
$*qFF$	-	-	-

- Nechť L je regulární jazyk. Rozhodněte a zdůvodněte, zda je jazyk $K = \{w \mid \#w\$ \in L\}$ regulární.
- Uvažujme konečný automat $A = (Q, \Sigma, \delta, q_0, F)$. Navrhněte dvousměrné automaty (pro L_1 a L_4 nedeterministické), které přijímají jazyky:
 - $L_1 = \{\#w\$ \mid ww^R \in L(A)\}$
 - $L_2 = \{\#w\$ \mid ww \in L(A)\}$
 - $L_3 = \{\#w\$ \mid (\exists v \in \Sigma^*) wv \in L(A) \& |w| = |v|\}$
 - $L_4 = \{\#w\$ \mid (\exists u, v \in \Sigma^*) w = uv \& uu^R v \in L(A)\}$
- Navrhněte nedeterministický konečný automat přijímající jazyk $L_1 = \{\#w\$ \mid ww^R \in L(A)\}$. Přitom nevyužívejte znalosti dvousměrných automatů.
- Popište princip konstrukce ekvivalentního konečného automatu k danému dvousměrnému automatu.
- Pro jazyky L_1, L_2, L_3, L_4 navrhněte (nedeterministické) konečné automaty, které je přijímají. Při návrhu využijte znalosti dvousměrných automatů.

10 Turingovy stroje

1. Navrhňte Turingův stroj $M = (Q, \{a, b\}, \Gamma, \delta, q_0, B, \{f_{yes}, f_{no}\})$, který zjistí, zda se ve vstupním slově vyskytuje písmeno a a skončí ve stavu f_{yes} nebo f_{no} .
2. Navrhňte Turingův stroj $M = (Q, \{a, b\}, \Gamma, \delta, q_0, B, \{f_a, f_b, f_{=}\})$, který zjistí, zda slovo obsahuje více a nebo b , skončí ve stavu f_a , f_b nebo $f_{=}$.
 - a) Můžete uvažovat vstupní slovo ve tvaru $a^i b^j$.
 - b) Totéž pro slovo z $\{a, b\}^*$.
 - c) Upravte přechozí stroj, aby smazal vstupní pásku a nechal na ní 'přebývajících' písmena napsaná bez mezer za sebou.
3. Navrhňte Turingův stroj $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$, který ze zadaného vstupního slova vytvoří jeho zrcadlový obraz. Přesněji, počáteční konfiguraci $q_0 w$ převede na $f w^R$, kde $f \in F$.
4. Navrhňte Turingův stroj $M = (Q, \Sigma = \{0, 1\}, \Gamma, \delta, q_0, B, F)$, který za znak $\#$ přidá tři nuly, zbytek slova odsune o tři pozice doprava. Počáteční konfiguraci $q_0 u \# v$, kde $u, v \in (\Sigma \setminus \#)^*$ převede na $f u \# 000 v$, kde $f \in F$.
5. Navrhňte Turingův stroj $M = (Q, \Sigma = \{0, 1\}, \Gamma, \delta, q_0, B, F)$, který prohodí obsah dvou paměťových bloků, přičemž paměť je představována páskou.
 - a) Počáteční konfiguraci $q_0 u \# v$, kde $u, v \in (\Sigma \setminus \#)^*$ převede na $f v \# u$, kde $f \in F$.
 - b) Počáteční konfiguraci $q_0 u \# v \# w \# x \# y$, kde $u, v, w, x, y \in (\Sigma \setminus \#)^*$ převede na $f u \# x \# w \# v \# y$, kde $f \in F$.
 - c) Vynasnažte se u předchozího, aby využíval co nejméně dalšího prostoru na pásce a co nejméně stavů.
6. Navrhňte Turingův stroj $M = (Q, \{0, 1, c\}, \Gamma, \delta, q_0, B, \{f_{yes}, f_{no}\})$, zkontroluje, jestli je vstupní slovo ve formátu $w c w$, $w \in \{0, 1\}^*$.
7. Navrhňte Turingův stroj $M = (Q, \{0, 1, c\}, \Gamma, \delta, q_0, B, \{f_{liche}, f_{sude}\})$, který najde střed slova a skončí buď na něm (lichá délka) nebo na prvním znaku druhé poloviny.
8. Navrhňte Turingův stroj $M = (Q, \{0, 1\}, \Gamma, \delta, q_0, B, \{f_{yes}, f_{no}\})$, zkontroluje, jestli je vstupní slovo ve formátu $w w$, $w \in \{0, 1\}^*$.
9. Naprogramujte na Turingově stroji *asociativní paměť*. Například můžete předpokládat Turingův stroj se dvěma páskami, kde první slouží k dotazování a odpovídání, a druhá reprezentuje obsah paměti.
 - a) Dotazem $k \# w$, kde $k, w \in (\Sigma \setminus \#)^*$ jsou po řadě klíč a data, vložíme do paměti asociaci klíče k s daty w . Vstupní páska je v tomto případě smazána, pokud klíč v paměti ještě není reprezentován, jinak je ponechána beze změny. Za oddělovač dvojic můžete zvolit nový symbol páskové abecedy.
 - b) Na první pásce lze položit dotaz: k , kde $k \in (\Sigma \setminus \#)^*$ je klíč, přičemž jako odpověď na první pásce očekáváme data asociované s klíčem k .
10. Necht $b \in \mathbb{N}$. Navrhňte **Turingův stroj**, který přijímá jazyk $L = \{u \# v \# w \mid u, v, w \in (0, 1, \dots, (b-1))^* \text{ \& } u_b \circ v_b = w_b\}$, u_b kde značí interpretaci slova u jako čísla v soustavě o základu b (tj. například $0101_2 = 5$) a $\circ$ nějakou binární aritmetickou operaci, kde $\circ \in \{+, -, *, \text{div}, \text{mod}\}$.
11. Necht $b \in \mathbb{N}$. Navrhňte **Turingovy stroje**, které přijímají jazyky:
 - (a) $L = \{w \mid w \in (0, 1, \dots, (b-1))^* \text{ \& } w_b \text{ je složené číslo}\}$
 - (b) $L = \{w \mid w \in (0, 1, \dots, (b-1))^* \text{ \& } w_b \text{ je prvočíslo}\}$.

11 Úvod do složitosti

1. Ukažte, že problémy clique, independent-set a vertex-cover, definované níže, jsou na sebe navzájem polynomiálně redukovatelné.
 - Clique
IN Graf $G = (V, E)$ a přirozené číslo $k \geq 0$.
Q Obsahuje G úplný podgraf na alespoň k vrcholech?
 - INDEPENDENT-SET
IN Graf $G = (V, E)$ a přirozené číslo $k \geq 0$.
Q Obsahuje G nezávislou množinu alespoň k vrcholů, tj. množinu $S \subseteq V$, $|S| \geq k$, kde žádné dva vrcholy nejsou spojeny hranou?
 - Vertex-Cover
IN Graf $G = (V, E)$ a přirozené číslo $k \geq 0$.
Q Obsahuje G vrcholové pokrytí velikosti nejvýše k , tj. množinu $S \subseteq V$, $|S| \leq k$, která má alespoň jeden vrchol z každé hrany?
2. Ukažte, že třída P je uzavřená na sjednocení, průnik a doplněk.
3. Ukažte, že třída NP je uzavřená na sjednocení a průnik.
4. Formulujte Cook-Levinovu větu a ideu jejího důkazu.