

## Zpracování přirozeného jazyka

Aleš Horák

E-mail: `hales@fi.muni.cz`

`http://nlp.fi.muni.cz/uui/`

Obsah:

- Komunikace
- Gramatiky
- Analýza přirozeného jazyka
- PA026 – Projekt z umělé inteligence

## PŘIROZENÝ JAZYK – PROSTŘEDEK KOMUNIKACE

**komunikace** = cílená výměna informace pomocí produkce a vnímání (sdílených) **pokynů**

- zvířata – až stovky pokynů (šimpanz, delfín, ...)
- člověk – potenciálně neomezené množství, díky přirozenému jazyku

## PŘIROZENÝ JAZYK – PROSTŘEDEK KOMUNIKACE

**komunikace** = cílená výměna informace pomocí produkce a vnímání (sdílených) **pokynů**

- zvířata – až stovky pokynů (šimpanz, delfín, ...)
- člověk – potenciálně neomezené množství, díky přirozenému jazyku

2 náhledy na **přirozený jazyk**:

- klasický (před 1953)** – jazyk se skládá z vět, které jsou buď pravdivé nebo nepravdivé (srovnej s logikou)
- moderní (po 1953)** – užití jazyka je jedna z možných **akcí**
  - Wittgenstein (1953) **Philosophical Investigations**
  - Searle (1969) **Speech Acts**

## PŘIROZENÝ JAZYK – PROSTŘEDEK KOMUNIKACE

**komunikace** = cílená výměna informace pomocí produkce a vnímání (sdílených) **pokynů**

- zvířata – až stovky pokynů (šimpanz, delfín, ...)
- člověk – potenciálně neomezené množství, díky přirozenému jazyku

2 náhledy na **přirozený jazyk**:

**klasický (před 1953)** – jazyk se skládá z vět, které jsou buď pravdivé nebo nepravdivé (srovnej s logikou)

**moderní (po 1953)** – užití jazyka je jedna z možných **akcí**

Wittgenstein (1953) **Philosophical Investigations**

Searle (1969) **Speech Acts**

Turingův test založen na jazyku  $\Leftarrow$  jazyk je pevně spojen s **myšlením**

komunikace se tvoří pomocí **řečových aktů** (*speech acts*) jako jeden z typů agentových akcí

**cíl** komunikace –

## PŘIROZENÝ JAZYK – PROSTŘEDEK KOMUNIKACE

**komunikace** = cílená výměna informace pomocí produkce a vnímání (sdílených) **pokynů**

- zvířata – až stovky pokynů (šimpanz, delfín, ...)
- člověk – potenciálně neomezené množství, díky přirozenému jazyku

2 náhledy na **přirozený jazyk**:

**klasický (před 1953)** – jazyk se skládá z vět, které jsou buď pravdivé nebo nepravdivé (srovnej s logikou)

**moderní (po 1953)** – užití jazyka je jedna z možných **akcí**

Wittgenstein (1953) **Philosophical Investigations**

Searle (1969) **Speech Acts**

Turingův test založen na jazyku  $\Leftarrow$  jazyk je pevně spojen s **myšlením**

komunikace se tvoří pomocí **řečových aktů** (*speech acts*) jako jeden z typů agentových akcí

**cíl** komunikace – **změnit** akce ostatních agentů

## ŘEČOVÉ AKTY

SITUACE

Mluvčí (*speaker*) → Promluva (*utterance*) → Posluchač (*hearer*)

řečové akty směřují k naplnění cílů mluvčího:

- |  |                                |
|--|--------------------------------|
| – informovat ( <i>inform</i> )                                 | “Před tebou je jáma.”          |
| – ptát se ( <i>query</i> )                                     | “Vidíš zlato?”                 |
| – přikázat/žádat ( <i>command/request</i> )                    | “Zvedni to.”                   |
| – slíbit/svěřit se s plánem ( <i>promise, commit to plan</i> ) | “Rozdělím se s tebou o zlato.” |
| – potvrdit ( <i>acknowledge</i> )                              | “OK”                           |

plánování řečových aktů vyžaduje znalosti:

- situace
- sémantiky a syntaxe (sdílených konvencí)
- informace o Posluchači – cíle, znalosti, rozumnost

## KOMUNIKAČNÍ FÁZE (PŘI INFORMOVÁNÍ)

průběh promluvy je možné rozložit na fáze:

- **záměr** (intention)  $M$  chce informovat  $P_o$ , že  $P_r$
- **generování** (generation)  $M$  vybírá slova  $W$  pro vyjádření  $P_r$
- **syntéza** (synthesis)  $M$  říká slova  $W$
  
- **vnímání** (perception)  $P_o$  vnímá  $W'$
- **analýza** (analysis)  $P_o$  odvozuje možné významy  $P_{r_1}, \dots, P_{r_n}$
- **zjednoznačnění** (disambiguation)  $P_o$  vybírá zamýšlený význam  $P_{r_i}$
- **zahrnutí** (incorporation)  $P_o$  zahrne  $P_{r_i}$  do své báze znalostí

## KOMUNIKAČNÍ FÁZE (PŘI INFORMOVÁNÍ)

průběh promluvy je možné rozložit na **fáze**:

- **záměr** (intention)  $M$  chce informovat  $P_o$ , že  $P_r$
- **generování** (generation)  $M$  vybírá slova  $W$  pro vyjádření  $P_r$
- **syntéza** (synthesis)  $M$  říká slova  $W$
  
- **vnímání** (perception)  $P_o$  vnímá  $W'$
- **analýza** (analysis)  $P_o$  odvozuje možné významy  $P_{r_1}, \dots, P_{r_n}$
- **zjednoznačnění** (disambiguation)  $P_o$  vybírá zamýšlený význam  $P_{r_i}$
- **zahrnutí** (incorporation)  $P_o$  zahrne  $P_{r_i}$  do své báze znalostí

Může přitom vzniknout **chyba**?

- neupřímnost ( $P_o$  nevěří  $P_r$ )
- víceznačnost promluvy ( $P_o$  zvolí špatné  $P_{r_i}$ )
- různé pochopení aktuální situace (zamýšlený význam mezi  $P_{r_i}$  není)



# KOMUNIKAČNÍ FÁZE – PŘÍKLAD

<p><b>záměr</b></p> <p><math>Vědět(P_0, \neg Na\_živu(Wumpus_1, S_3))</math></p>	<p><b>generování</b></p> <p>“Wumpus je mrtvý.”</p>	<p><b>syntéza</b> <span style="color: red;">MLUVČÍ</span></p> <p>[v u m p u s j e m r t v í:]</p>
--	--	---

<p><b>vnímání</b></p> <p>“Wumpus je mrtvý.”</p>	<p><b>analýza</b></p> <div style="text-align: center;"> <pre> graph TD     S --&gt; NP     S --&gt; VP     NP --&gt; Noun     VP --&gt; Verb     VP --&gt; Adjective     Noun --&gt; Wumpus     Verb --&gt; je     Adjective --&gt; mrtvý             </pre> </div> <p>syntaktická analýza:</p> <p>sémantická interpretace: <math>\neg Na\_živu(Wumpus, Teď)</math>  <math>Unavený(Wumpus, Teď)</math></p> <p>pragmatická interpretace: <math>\neg Na\_živu(Wumpus_1, S_3)</math>  <math>Unavený(Wumpus_1, S_3)</math></p>	<p><b>zjednoznačnění</b> <span style="color: red;">POSLUCHAČ</span></p> <p><math>\neg Na\_živu(Wumpus_1, S_3))</math></p> <hr/> <p><b>zahrnutí</b></p> <p><math>Tell(KB, \neg Na\_živu(Wumpus_1, S_3))</math></p>
---	--	---

---

---

✓ ●	Komunikace . . . . .	2
⇒ ●	Gramatiky . . . . .	6
●	Analýza přirozeného jazyka . . . . .	25
●	PA026 – Projekt z umělé inteligence . . . . .	33

## GRAMATIKY

zvířata používají místo vět izolované symboly  $\Rightarrow$  **omezená** sada komunikovatelných situací  
 $\rightarrow$  žádná **generativní kapacita**

**gramatika** specifikuje skladební strukturu složených pokynů – definuje *formální jazyk* pokynů

**formální jazyk** = množina **řetězců** (vět) **terminálních symbolů** (slov)

2 náhledy na vztah věty a gramatiky:

- $S$  je správný řetězec/věta z jazyka  $\Leftrightarrow S$  je **analyzovatelný** příslušnou gramatikou
- příslušná gramatika **generuje**  $S$   $\Leftrightarrow S$  je správný řetězec/věta z jazyka

gramatika je zadána jako množina **přepisovacích pravidel**, např.

$$S \rightarrow NP VP$$
$$Pronoun \rightarrow \text{já} \mid \text{ty} \mid \text{on} \mid \dots$$

v tomto příkladu:

$S$	<b>větný symbol</b> – kořenový symbol gramatiky
$NP, VP$	<b>neterminály</b>
$\text{já, ty, } \dots$	<b>terminály</b>

## TYPY GRAMATIK

gramatiky:

☐ **regulární** (regular)    **neterminál**  $\rightarrow$  **terminál**[neterminál]

$S \rightarrow aS$

$S \rightarrow b$

ekvivalentní síle **konečných automatů**, neumí  $a^n b^n$

☐ **bezkontextové** (context-free)    **neterminál**  $\rightarrow$  **cokoliv**

$S \rightarrow aSb$

ekvivalentní síle **zásobníkových automatů**, umí  $a^n b^n$ , neumí  $a^n b^n c^n$

☐ **kontextové** (context-sensitive) – víc neterminálů na levé straně; na levé straně se jejich počet “zmenšuje”

$ASB \rightarrow AAaBB$

umí  $a^n b^n c^n$

☐ **rekurzivně vyčíslitelné** (recursively enumerable) – bez omezení

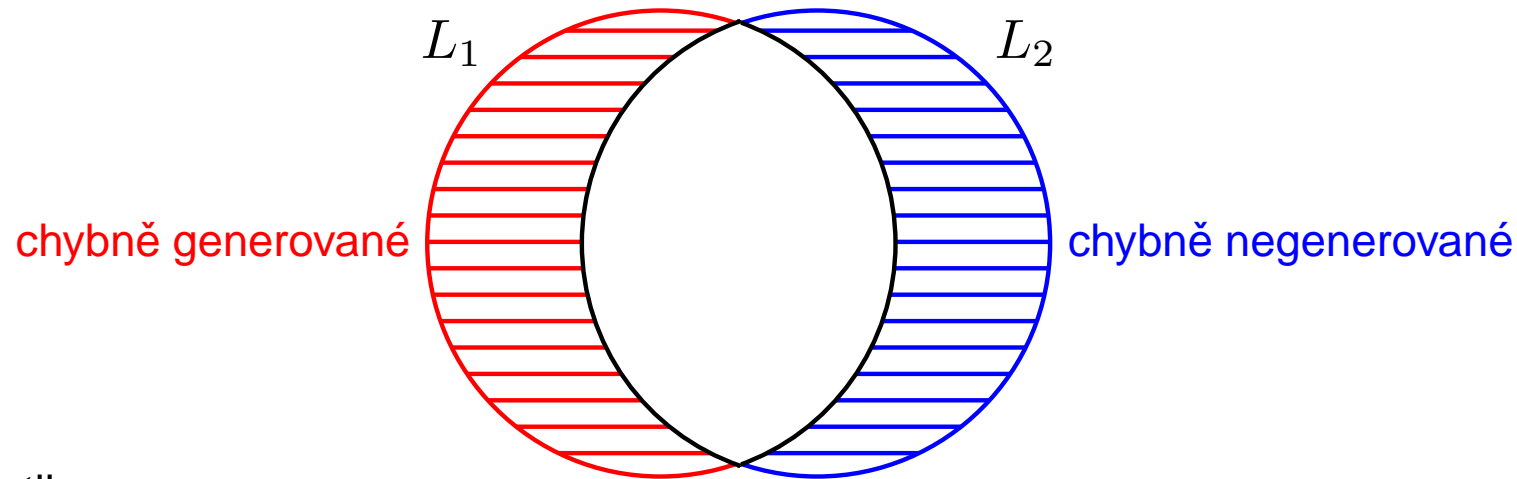
ekvivalentní síle **Turingova stroje**

**přirozený jazyk** byl dlouho pokládán za bezkontextový  $\rightarrow$  nyní prokázáno, že obsahuje **kontextové prvky**

## PŘESNOST A POKRYTÍ GRAMATIKY

u složitějších jazyků (např. přirozených)

→ jazyk  $L_1$  (generovaný gramatikou) se liší od zamýšleného jazyka  $L_2$



kvalita gramatiky:

- **pokrytí** – procento vět jazyka  $L_2$  generovatelných gramatikou ( $|L_1 \cap L_2|/|L_2|$ )
- **přesnost** – procento generovaných vět, které jsou správné věty jazyka  $L_2$  ( $|L_1 \cap L_2|/|L_1|$ )

tvorba gramatiky ... postupný proces zvyšování pokrytí a přesnosti

gramatiky přirozených jazyků – velmi rozsáhlé a přesto většinou nepopisují plně ani angličtinu ☹

## DC GRAMATIKY – GRAMATIKY USPOŘÁDANÝCH KLAUZULÍ

- *Definite-Clause Grammars*, **DCG**
- významná aplikace Prologu – *syntaktická analýza*
- DCG jsou **rozšířením bezkontextových gramatik** (CFG)
- jejich implementace využívá *rozdílových seznamů*

## DC GRAMATIKY – GRAMATIKY USPOŘÁDANÝCH KLAUZULÍ

- *Definite-Clause Grammars*, **DCG**
- významná aplikace Prologu – *syntaktická analýza*
- DCG jsou **rozšířením bezkontextových gramatik** (CFG)
- jejich implementace využívá *rozdílových seznamů*

### Formální podobnosti mezi DCG a CFG:

- CFG: pravidla tvaru  $x \rightarrow y$ , kde  $x \in N$  je neterminál a  $y \in (N \cup T)^*$  je konečná posloupnost terminálů a neterminálů
- DCG: pravidla tvaru  $\langle \mathbf{hlava} \rangle \dashrightarrow \langle \mathbf{tělo} \rangle$ , kde  $\langle \mathbf{hlava} \rangle$  je opět neterminál a  $\langle \mathbf{tělo} \rangle$  je opět konečná posloupnost terminálů a neterminálů
- pravidlo  $\langle \mathbf{hlava} \rangle \dashrightarrow \langle \mathbf{tělo} \rangle$  znamená, že jedním z možných tvarů  $\langle \mathbf{hlavy} \rangle$  je  $\mathbf{tělo}$ , neboli:  $\langle \mathbf{hlavu} \rangle$  je možno přepsat na  $\langle \mathbf{tělo} \rangle$

## ROZDÍLY A ROZŠÍŘENÍ DCG OPROTI CFG

1. **Neterminál** může být téměř libovolný term, kromě *seznamu*, *proměnné* a *čísla*.
2. **Terminál** může být libovolný term, s tím, že terminály a posloupnosti terminálů uzavíráme do hranatých závorek – jako **seznamy**.
3. Pravá strana pravidla může obsahovat **dodatečné podmínky** v podobě prologovských podcílů. Tyto podmínky uzavíráme do složených závorek.
4. Levá strana pravidla může dokonce vypadat i tak, že neterminál je následován posloupností terminálů.
5. Tělo pravidla smí obsahovat řez.



## DC GRAMATIKA – PŘÍKLAD 1

gramatika vět typu “The young boy sings a song.”

*% 1. část -- pravidla*

sentence --> noun\_phrase, verb\_phrase.

noun\_phrase --> determiner, noun\_phrase2.

noun\_phrase --> noun\_phrase2.

noun\_phrase2 --> adjective, noun\_phrase2.

noun\_phrase2 --> noun.

verb\_phrase --> verb.

verb\_phrase --> verb, noun\_phrase.

*% 2. část -- lexikon*

determiner --> [the].

noun --> [boy].

determiner --> [a].

noun --> [song].

verb --> [sings].

adjective --> [young].

## ANALÝZA V PROLOGU POMOCÍ APPEND

- větu reprezentujeme seznamem slov **[the,young,boy,sings,a,song]**
- **pravidlová část** – neterminál chápeme jako unární predikát, jehož argumentem je ta větná složka, kterou daný neterminál popisuje

```
sentence(S) :- append(NP,VP,S),  
               noun_phrase(NP), verb_phrase(VP).  
...
```

- **slovníková část, lexikon** – zapisujeme pomocí faktů:

```
determiner([the ]).      noun([boy]).  
determiner([a ]).      ...
```

## EFEKTIVNĚJI – ROZDÍLOVÉ SEZNAMY

přepis gramatiky do Prologu pomocí rozdílových seznamů:

```
sentence(S,S0) :- noun_phrase(S,S1), verb_phrase(S1,S0).  
  
noun_phrase(S,S0) :- determiner(S,S1), noun_phrase2(S1,S0).  
noun_phrase(S,S0) :- noun_phrase2(S,S0).  
noun_phrase2(S,S0) :- adjective(S,S1), noun_phrase2(S1,S0).  
noun_phrase2(S,S0) :- noun(S,S0).  
verb_phrase(S,S0) :- verb(S,S0).  
verb_phrase(S,S0) :- verb(S,S1), noun_phrase(S1,S0).  
  
determiner([the|S],S).      noun([boy|S],S).  
determiner([a|S],S).      noun([song|S],S).  
verb([sings|S],S).       adjective([young|S],S).
```

## EFEKTIVNĚJI – ROZDÍLOVÉ SEZNAMY

přepis gramatiky do Prologu pomocí **rozdílových seznamů**:

```
sentence(S,S0) :- noun_phrase(S,S1), verb_phrase(S1,S0).  
  
noun_phrase(S,S0) :- determiner(S,S1), noun_phrase2(S1,S0).  
noun_phrase(S,S0) :- noun_phrase2(S,S0).  
noun_phrase2(S,S0) :- adjective(S,S1), noun_phrase2(S1,S0).  
noun_phrase2(S,S0) :- noun(S,S0).  
verb_phrase(S,S0) :- verb(S,S0).  
verb_phrase(S,S0) :- verb(S,S1), noun_phrase(S1,S0).  
  
determiner([the|S],S).      noun([boy|S],S).  
determiner([a|S],S).      noun([song|S],S).  
verb([sings|S],S).        adjective([young|S],S).  
  
?- sentence([the,young,boy,sings,a,song],[]).  
Yes
```

## LEXIKON PRO AGENTA VE WUMPUSOVĚ JESKYNI

Gramatika přímo na slovech je příliš rozsáhlá. Řešením je rozdělení slov do **kategorií**:

podst. jméno: *Noun* → zápach | vánek | třpyt | nic | wumpuse | jáma | zlato | ...

sloveso: *Verb* → jsem | je | vidím | cítím | působí | zapáchá | jdu | ...

příd. jméno: *Adjective* → levý | pravý | východní | jižní | ...

příslovce: *Adverb* → tady | tam | blízko | vpředu | vpravo | vlevo | východně | jižně  
| vzadu | ...

vl. jméno: *Name* → Petr | Honza | Brno | FI MU | ...

zájmeno: *Pronoun* → já | ty | mě | toho | ten | ta ...

předložka: *Preposition* → do | v | na | u | ...

spojka: *Conjunction* → a | nebo | ale | ...

číslice: *Digit* → 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

kategorie můžeme dělit na **otevřené** (vyvíjející se) a **uzavřené** (stálé)

## MORFOLOGICKÁ ANALÝZA

- V češtině u lexikonu nestačí prostý výčet tvarů – je nutná **morfologická analýza** (morfologie=tvarosloví)
- skloňovaná a časovaná slova se rozkládají na **segmenty**

*pří-lež-it-ost-n-ými*

*pří* – prefix; *lež* – kořen; *it, ost, n* – suffixy; *ými* – koncovka

## MORFOLOGICKÁ ANALÝZA

- V češtině u lexikonu nestačí prostý výčet tvarů – je nutná **morfologická analýza** (morfologie=tvarosloví)
- skloňovaná a časovaná slova se rozkládají na **segmenty**

*pří-lež-it-ost-n-ými*

*pří* – prefix; *lež* – kořen; *it, ost, n* – suffixy; *ými* – koncovka

- každé slovo má **základní tvar** (*lemma*), podle koncovky se určují **gramatické kategorie**

*% slovník základních gramatických kategorií – – pád, číslo, rod*

*% adj(+Slovo, +Lemma, +Pád, +Císlo, +Rod)*

**adj**(chytrý, chytrý, 1, sg, mz).      **adj**(chytrého, chytrý, 2, sg, mz).      **adj**( chytří , chytrý, 1, pl, mz).

## MORFOLOGICKÁ ANALÝZA

- V češtině u lexikonu nestačí prostý výčet tvarů – je nutná **morfologická analýza** (morfologie=tvarosloví)
- skloňovaná a časovaná slova se rozkládají na **segmenty**

*pří-lež-it-ost-n-ými*

*pří* – prefix; *lež* – kořen; *it, ost, n* – suffixy; *ými* – koncovka

- každé slovo má **základní tvar** (*lemma*), podle koncovky se určují **gramatické kategorie**

*% slovník základních gramatických kategorií – – pád, číslo, rod*

*% adj(+Slovo, +Lemma, +Pád, +Císlo, +Rod)*

*adj(chytrý, chytrý, 1, sg, mz). adj(chytrého, chytrý, 2, sg, mz). adj(chytří, chytrý, 1, pl, mz).*

- reálná morfologická analýza ČJ – program AJKA na FI MU

<http://nlp.fi.muni.cz/projekty/wwwajka/>

```
ajka>nejneuvěřitelněji
```

```
<s> nej-ne=uvěřiteln==ěji= (1022)
```

```
<l>uvěřitelně
```

```
<c>k6xMeNd3
```

```
ajka>hnát
```

```
<s> ==hná=t= (618)
```

```
<l>hnát
```

```
<c>k5eAmFaI
```

```
<s> =hnát=== (1030)
```

```
<l>hnát
```

```
<c>k1gInSc1,k1gInSc4
```



## GRAMATICKÁ PRAVIDLA PRO AGENTA VE WUMPUSOVĚ JESKYNI

S	→	<i>NP VP</i>	%	já + cítím vánek
		<i>S Conjunction S</i>	%	já cítím vánek + a + já jdu na východ
NP	→	<i>Pronoun</i>	%	já
		<i>Noun</i>	%	jáma
		<i>Adjective Noun</i>	%	levá jáma
		<i>Pronoun NP</i>	%	toho + wumpuse
		<i>Noun Digit ‘,’ Digit</i>	%	pole + 3,4
		<i>NP PP</i>	%	jáma + na východě
		<i>NP RelClause</i>	%	toho wumpuse + ,který zapáchá
VP	→	<i>Verb</i>	%	zapáchá
		<i>VP NP</i>	%	cítím + vánek
		<i>VP Adjective</i>	%	je + třpytivý
		<i>VP PP</i>	%	jdu + na východ
		<i>VP Adverb   Adverb VP</i>	%	jdu + dopředu
PP	→	<i>Preposition NP</i>	%	na + východ
RelClause	→	<i>‘, který’ VP</i>	%	,který + zapáchá

## SYNTAKTICKÝ STROM

**syntaktický strom** vzniká během **syntaktické analýzy** a dává **záznam** o jejím průběhu:

# SYNTAKTICKÝ STROM

**syntaktický strom** vzniká během **syntaktické analýzy** a dává **záznam** o jejím průběhu:

Východní

jáma

tady

působí

vánek

# SYNTAKTICKÝ STROM

**syntaktický strom** vzniká během **syntaktické analýzy** a dává **záznam** o jejím průběhu:

*Adjective*



Východní

*Noun*



jáma

*Adverb*



tady

*Verb*



působí

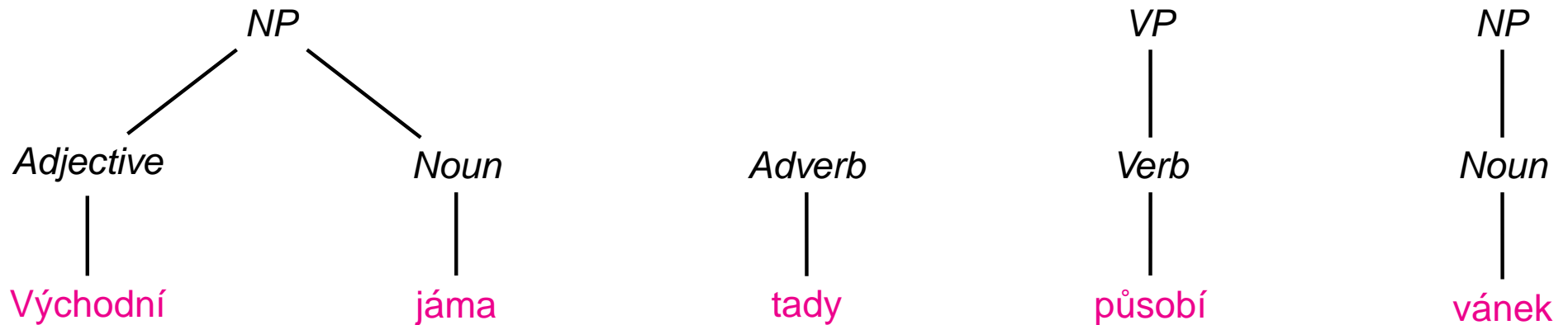
*Noun*



vánek

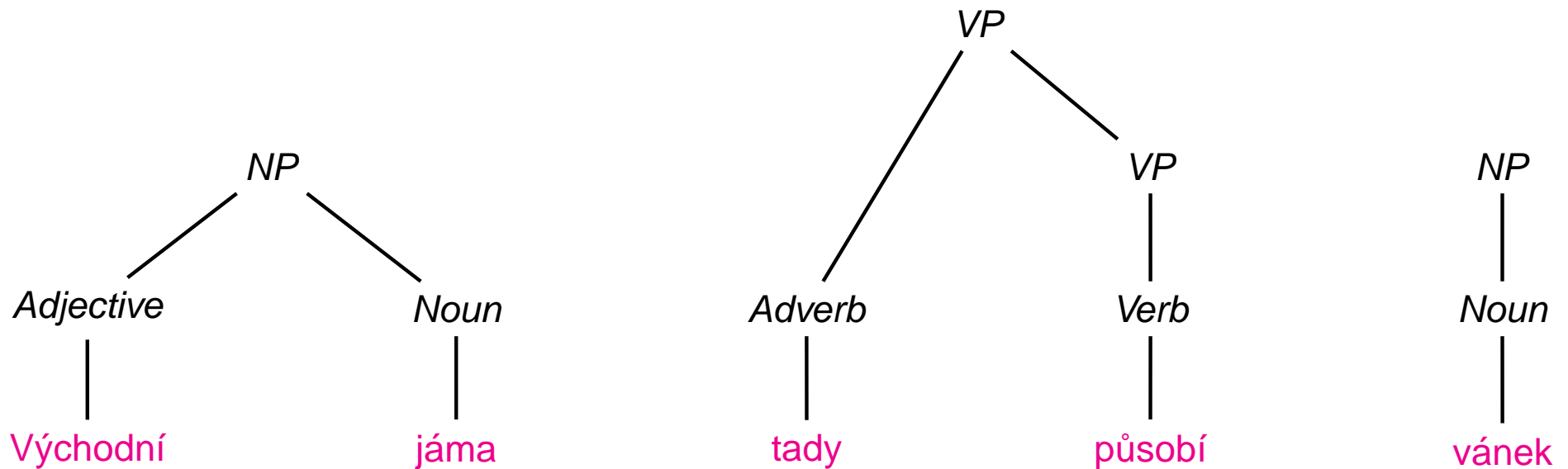
## SYNTAKTICKÝ STROM

syntaktický strom vzniká během syntaktické analýzy a dává záznam o jejím průběhu:



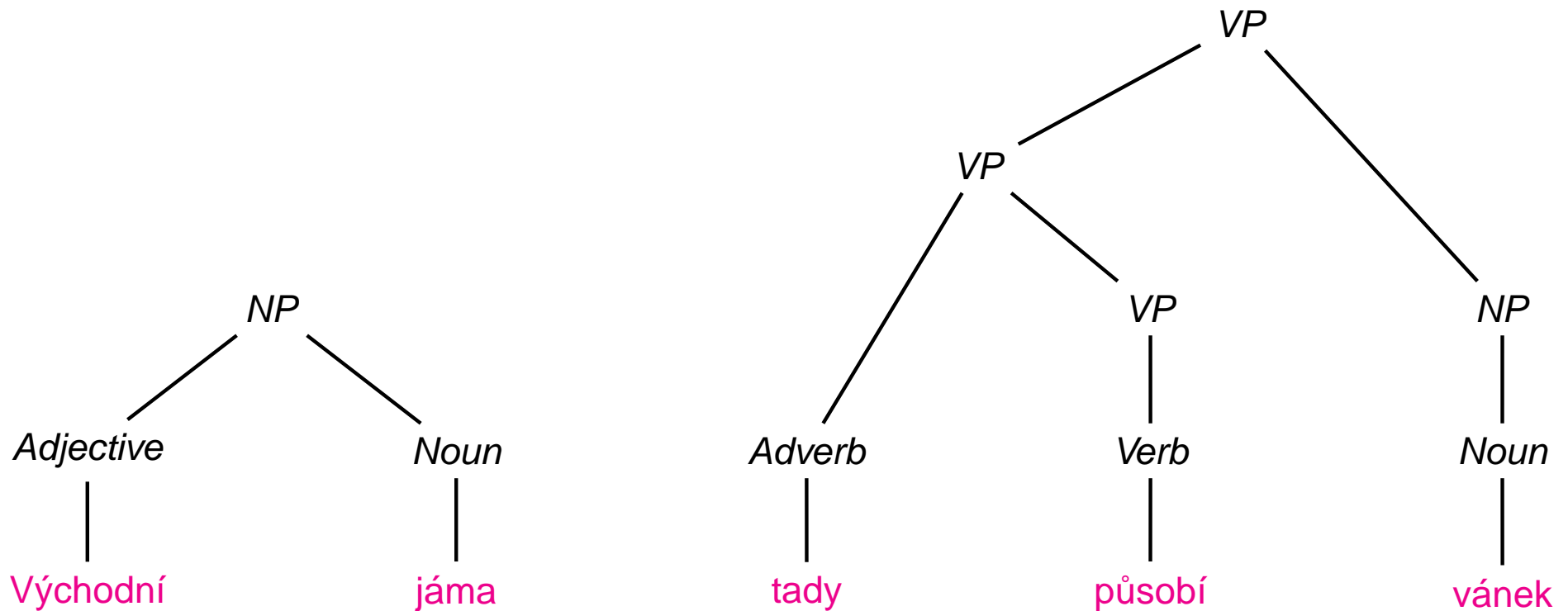
## SYNTAKTICKÝ STROM

syntaktický strom vzniká během syntaktické analýzy a dává záznam o jejím průběhu:



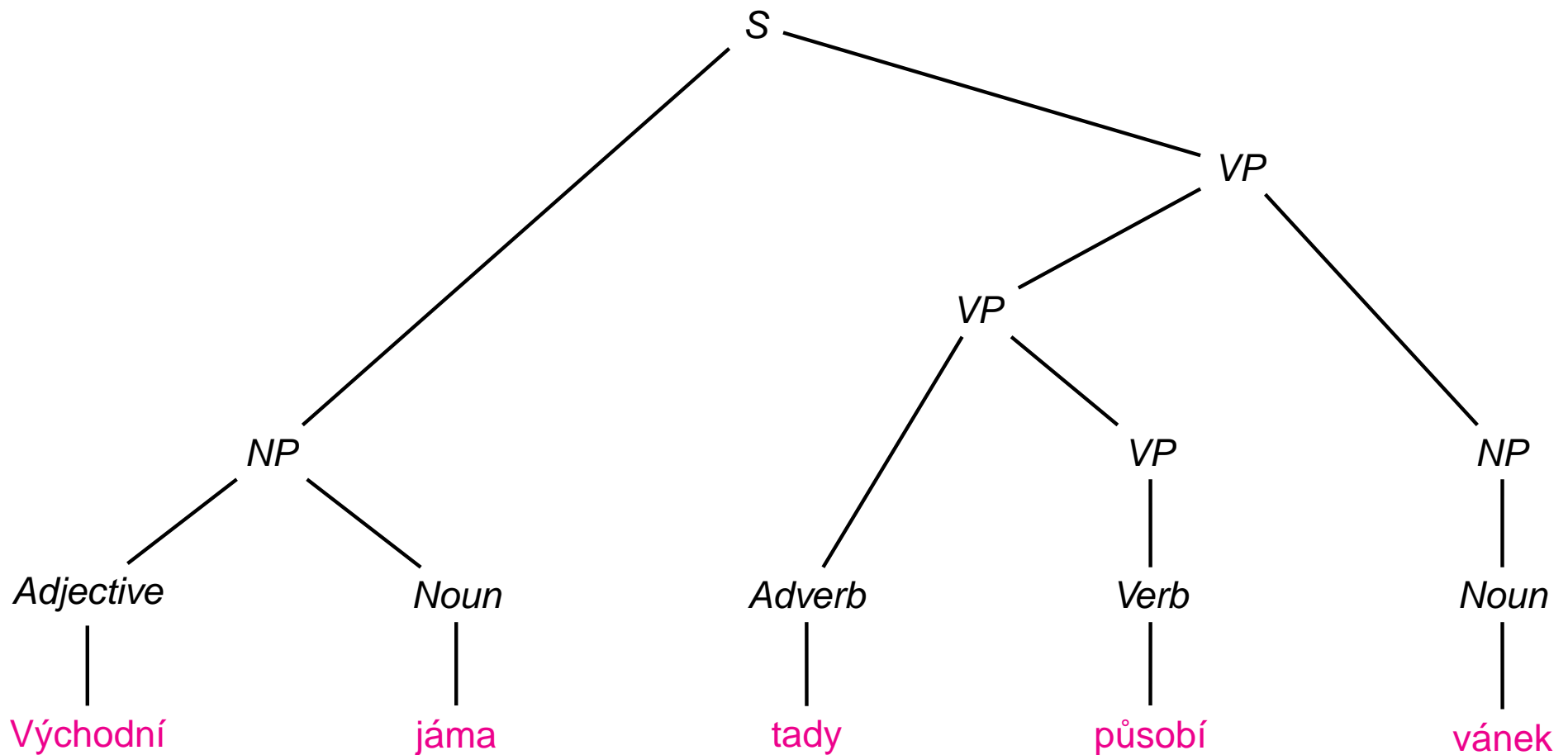
## SYNTAKTICKÝ STROM

syntaktický strom vzniká během syntaktické analýzy a dává záznam o jejím průběhu:



## SYNTAKTICKÝ STROM

syntaktický strom vzniká během syntaktické analýzy a dává záznam o jejím průběhu:





## KONSTRUKCE DERIVAČNÍHO STROMU

Neterminály opatříme argumentem:

```
sentence(sentence(NP,VP)) --> noun_phrase(NP), verb_phrase(VP).
```

Převod do podoby klauzulí:

```
sentence(sentence(NP,VP),S,S0) :- noun_phrase(NP,S,S1), verb_phrase(VP,S1,S0).
```

## DC GRAMATIKA S KONSTRUKCÍ STROMU ANALÝZY

sentence(**s**(N,V)) --> noun\_phrase(N), verb\_phrase(V).  
noun\_phrase(**np**(D,N)) --> determiner(D), noun\_phrase2(N).  
noun\_phrase(**np**(N)) --> noun\_phrase2(N).  
noun\_phrase2(**np2**(A,N)) --> adjective(A), noun\_phrase2(N).  
noun\_phrase2(**np2**(N)) --> noun(N).  
verb\_phrase(**vp**(V)) --> verb(V).  
verb\_phrase(**vp**(V,N)) --> verb(V), noun\_phrase(N).

determiner(**det**(the)) --> [the].  
determiner(**det**(a)) --> [a].  
adjective(**adj**(young)) --> [young].  
noun(**noun**(boy)) --> [boy].  
noun(**noun**(song)) --> [song].  
verb(**verb**(sings)) --> [sings].

## DC GRAMATIKA S KONSTRUKCÍ STROMU ANALÝZY

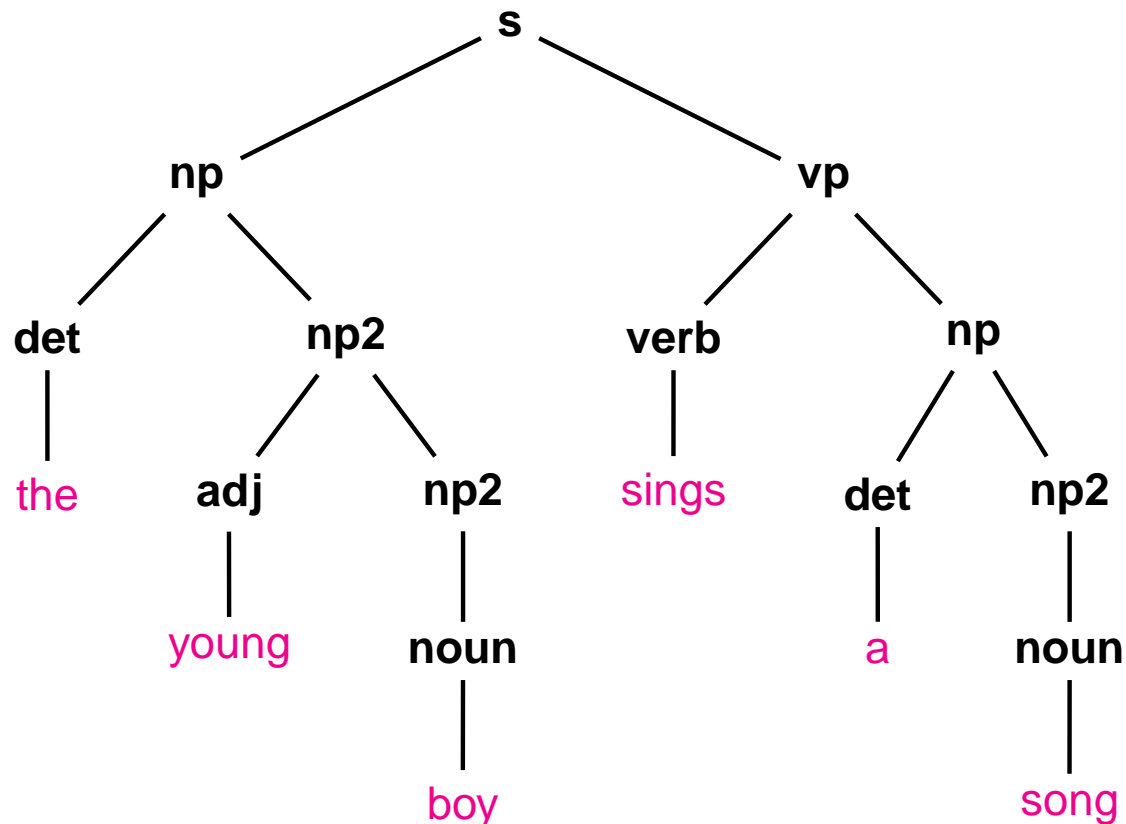
`sentence(s(N,V))`  $\rightarrow$  `noun_phrase(N)`, `verb_phrase(V)`.  
`noun_phrase(np(D,N))`  $\rightarrow$  `determiner(D)`, `noun_phrase2(N)`.  
`noun_phrase(np(N))`  $\rightarrow$  `noun_phrase2(N)`.  
`noun_phrase2(np2(A,N))`  $\rightarrow$  `adjective(A)`, `noun_phrase2(N)`.  
`noun_phrase2(np2(N))`  $\rightarrow$  `noun(N)`.  
`verb_phrase(vp(V))`  $\rightarrow$  `verb(V)`.  
`verb_phrase(vp(V,N))`  $\rightarrow$  `verb(V)`, `noun_phrase(N)`.

`determiner(det(the))`  $\rightarrow$  [the].  
`determiner(det(a))`  $\rightarrow$  [a].  
`adjective(adj(young))`  $\rightarrow$  [young].  
`noun(noun(boy))`  $\rightarrow$  [boy].  
`noun(noun(song))`  $\rightarrow$  [song].  
`verb(verb(sings))`  $\rightarrow$  [sings].

?– `sentence(Tree, [the,young,boy,sings,a,song],[[]])`.  
Tree=`s(np(det(the),np2(adj(young),np2(noun(boy))))),`  
    `vp(verb(sings),np(det(a),np2(noun(song))))`)

## DERIVAČNÍ STROM ANALÝZY V DC GRAMATIKÁCH

? – `sentence(Tree, [the, young, boy, sings, a, song], [])`.  
Tree=`s(np(det(the), np2(adj(young), np2(noun(boy))))),`  
`vp(verb(sings), np(det(a), np2(noun(song))))`)



## TEST NA SHODU

Pokud však rozšíříme slovník:

`noun(noun(boys)) --> [boys].`  
`verb(verb(sing)) --> [sing].`

Narazíme na problém se shodou v čísle:

?– `sentence(.,[a, young, boys, sings ],[]).`  
Yes

?– `sentence(.,[a, boy, sing ],[]).`  
Yes

## TEST NA SHODU

Pokud však rozšíříme slovník:

`noun(noun(boys)) --> [boys].`  
`verb(verb(sing)) --> [sing].`

Narazíme na problém se shodou v čísle:

?– `sentence(.,[a, young, boys, sings ],[]).`  
Yes

?– `sentence(.,[a, boy, sing ],[]).`  
Yes

Proto rozšíříme neterminály o další argument **Num**, ve kterém můžeme testovat shodu:

`sentence(sentence(NP,VP)) --> noun_phrase(NP,Num), verb_phrase(VP,Num).`

## DC GRAMATIKA S TESTY NA SHODU

`sentence(sentence(N,V))` --> `noun_phrase(N,Num)`, `verb_phrase(V,Num)`.  
`noun_phrase(np(D,N),Num)` --> `determiner(D,Num)`, `noun_phrase2(N,Num)`.  
`noun_phrase(np(N),Num)` --> `noun_phrase2(N,Num)`.  
`noun_phrase2(np2(A,N),Num)` --> `adjective(A)`, `noun_phrase2(N,Num)`.  
`noun_phrase2(np2(N),Num)` --> `noun(N,Num)`.  
`verb_phrase(vp(V),Num)` --> `verb(V,Num)`.  
`verb_phrase(vp(V,N),Num)` --> `verb(V,Num)`, `noun_phrase(N,Num1)`.

`determiner(det(the),_)` --> [the].  
`determiner(det(a),sg)` --> [a].  
`verb(verb(sings),sg)` --> [sings].  
`verb(verb(sing),pl)` --> [sing].  
`adjective(adj(young))` --> [young].

`noun(noun(boy),sg)` --> [boy].  
`noun(noun(song),sg)` --> [song].  
`noun(noun(boys),pl)` --> [boys].  
`noun(noun(songs),pl)` --> [songs].

## DC GRAMATIKA S TESTY NA SHODU

`sentence(sentence(N,V))` --> `noun_phrase(N,Num)`, `verb_phrase(V,Num)`.  
`noun_phrase(np(D,N),Num)` --> `determiner(D,Num)`, `noun_phrase2(N,Num)`.  
`noun_phrase(np(N),Num)` --> `noun_phrase2(N,Num)`.  
`noun_phrase2(np2(A,N),Num)` --> `adjective(A)`, `noun_phrase2(N,Num)`.  
`noun_phrase2(np2(N),Num)` --> `noun(N,Num)`.  
`verb_phrase(vp(V),Num)` --> `verb(V,Num)`.  
`verb_phrase(vp(V,N),Num)` --> `verb(V,Num)`, `noun_phrase(N,Num1)`.

<code>determiner(det(the),_)</code> --> [the].	<code>noun(noun(boy),sg)</code> --> [boy].
<code>determiner(det(a),sg)</code> --> [a].	<code>noun(noun(song),sg)</code> --> [song].
<code>verb(verb(sings),sg)</code> --> [sings].	<code>noun(noun(boys),pl)</code> --> [boys].
<code>verb(verb(sing),pl)</code> --> [sing].	<code>noun(noun(songs),pl)</code> --> [songs].
<code>adjective(adj(young))</code> --> [young].	

?– `sentence(_, [a, young, boys, sings ], [])`.

No



## DC GRAMATIKA S TESTY NA SHODU

`sentence(sentence(N,V))` --> `noun_phrase(N,Num)`, `verb_phrase(V,Num)`.  
`noun_phrase(np(D,N),Num)` --> `determiner(D,Num)`, `noun_phrase2(N,Num)`.  
`noun_phrase(np(N),Num)` --> `noun_phrase2(N,Num)`.  
`noun_phrase2(np2(A,N),Num)` --> `adjective(A)`, `noun_phrase2(N,Num)`.  
`noun_phrase2(np2(N),Num)` --> `noun(N,Num)`.  
`verb_phrase(vp(V),Num)` --> `verb(V,Num)`.  
`verb_phrase(vp(V,N),Num)` --> `verb(V,Num)`, `noun_phrase(N,Num1)`.

<code>determiner(det(the),_)</code> --> [the].	<code>noun(noun(boy),sg)</code> --> [boy].
<code>determiner(det(a),sg)</code> --> [a].	<code>noun(noun(song),sg)</code> --> [song].
<code>verb(verb(sings),sg)</code> --> [sings].	<code>noun(noun(boys),pl)</code> --> [boys].
<code>verb(verb(sing),pl)</code> --> [sing].	<code>noun(noun(songs),pl)</code> --> [songs].
<code>adjective(adj(young))</code> --> [young].	

?– `sentence(,[a, young, boys, sings ],[ ])`.

No

?– `sentence(,[the,boys,sings,a,song ],[ ])`.

No

## DC GRAMATIKA S TESTY NA SHODU

sentence(sentence(N,V)) --> noun\_phrase(N,Num), verb\_phrase(V,Num).  
 noun\_phrase(np(D,N),Num) --> determiner(D,Num), noun\_phrase2(N,Num).  
 noun\_phrase(np(N),Num) --> noun\_phrase2(N,Num).  
 noun\_phrase2(np2(A,N),Num) --> adjective(A), noun\_phrase2(N,Num).  
 noun\_phrase2(np2(N),Num) --> noun(N,Num).  
 verb\_phrase(vp(V),Num) --> verb(V,Num).  
 verb\_phrase(vp(V,N),Num) --> verb(V,Num), noun\_phrase(N,Num1).

determiner(det(the),_) --> [the].	noun(noun(boy),sg) --> [boy].
determiner(det(a),sg) --> [a].	noun(noun(song),sg) --> [song].
verb(verb(sings),sg) --> [sings].	noun(noun(boys),pl) --> [boys].
verb(verb(sing),pl) --> [sing].	noun(noun(songs),pl) --> [songs].
adjective(adj(young)) --> [young].	

?– sentence(\_, [a, young, boys, sings ], []).

No

?– sentence(\_, [the, boys, sings, a, song ], []).

No

?– sentence(\_, [the, boys, sing, a, song ], []).

Yes

## PODMÍNKY V TĚLE PRAVIDEL

DC gramatiky mohou mít pomocné **podmínky** v těle pravidel – libovolný **Prologovský kód**

např. CFG pro vyhodnocení aritmetického výrazu:

$$\begin{aligned} E &\rightarrow T + E \mid T - E \mid T \\ T &\rightarrow F * T \mid F / T \mid F \\ F &\rightarrow (E) \mid f \end{aligned}$$

zapišeme **včetně výpočtu** hodnoty výrazu:

```
expr(X) ---> term(Y), [+], expr(Z), {X is Y+Z}.
expr(X) ---> term(Y), [-], expr(Z), {X is Y-Z}.
expr(X) ---> term(X).

term(X) ---> factor(Y), [*], term(Z), {X is Y*Z}.
term(X) ---> factor(Y), [/], term(Z), {X is Y/Z}.
term(X) ---> factor(X).

factor(X) ---> ['(', expr(X), [')'].
factor(X) ---> [X], {integer(X)}.
```

## PODMÍNKY V TĚLE PRAVIDEL

DC gramatiky mohou mít pomocné **podmínky** v těle pravidel – libovolný **Prologovský kód**

např. CFG pro vyhodnocení aritmetického výrazu:

$$E \rightarrow T + E \mid T - E \mid T$$

$$T \rightarrow F * T \mid F / T \mid F$$

$$F \rightarrow (E) \mid f$$

zapišeme **včetně výpočtu** hodnoty výrazu:

`expr(X) ---> term(Y), [+], expr(Z), {X is Y+Z}.`

`expr(X) ---> term(Y), [-], expr(Z), {X is Y-Z}.`

`expr(X) ---> term(X).`

`term(X) ---> factor(Y), [*], term(Z), {X is Y*Z}.`

`term(X) ---> factor(Y), [/], term(Z), {X is Y/Z}.`

`term(X) ---> factor(X).`

`factor(X) ---> ['(', expr(X), [')']]`

`factor(X) ---> [X], {integer(X)}.`

?- `expr(X,[3,+4,/2,-,'(' ,2,*6,/3,+2, ')'] ,[]).`

`X = -1`

# GENERATIVNÍ SÍLA DCG

Generativní (rozpoznávací) síla DCG je větší než CFG

např. jazyk  $a^n b^n c^n$ :

$abc \longrightarrow a(N), b(N), c(N).$

$a(0) \longrightarrow [].$

$a(s(N)) \longrightarrow [a], a(N).$

$b(0) \longrightarrow [].$

$b(s(N)) \longrightarrow [b], b(N).$

$c(0) \longrightarrow [].$

$c(s(N)) \longrightarrow [c], c(N).$

## GENERATIVNÍ SÍLA DCG

Generativní (rozpoznávací) síla DCG je větší než CFG

např. jazyk  $a^n b^n c^n$ :

$abc \text{ ---} > a(N), b(N), c(N).$

$a(0) \text{ ---} > [].$

$a(s(N)) \text{ ---} > [a], a(N).$

$b(0) \text{ ---} > [].$

$b(s(N)) \text{ ---} > [b], b(N).$

$c(0) \text{ ---} > [].$

$c(s(N)) \text{ ---} > [c], c(N).$

? –  $abc(X, []).$

## GENERATIVNÍ SÍLA DCG

Generativní (rozpoznávací) síla DCG je větší než CFG

např. jazyk  $a^n b^n c^n$ :

$abc \text{ ---} > a(N), b(N), c(N).$

$a(0) \text{ ---} > [].$

$a(s(N)) \text{ ---} > [a], a(N).$

$b(0) \text{ ---} > [].$

$b(s(N)) \text{ ---} > [b], b(N).$

$c(0) \text{ ---} > [].$

$c(s(N)) \text{ ---} > [c], c(N).$

?-  $abc(X, []).$

$X = [] ;$

## GENERATIVNÍ SÍLA DCG

Generativní (rozpoznávací) síla DCG je větší než CFG

např. jazyk  $a^n b^n c^n$ :

$abc \text{ ---} > a(N), b(N), c(N).$

$a(0) \text{ ---} > [].$

$a(s(N)) \text{ ---} > [a], a(N).$

$b(0) \text{ ---} > [].$

$b(s(N)) \text{ ---} > [b], b(N).$

$c(0) \text{ ---} > [].$

$c(s(N)) \text{ ---} > [c], c(N).$

?-  $abc(X, []).$

$X = [] ;$

$X = [a, b, c] ;$



## GENERATIVNÍ SÍLA DCG

Generativní (rozpoznávací) síla DCG je větší než CFG

např. jazyk  $a^n b^n c^n$ :

$abc \text{ ---} > a(N), b(N), c(N).$

$a(0) \text{ ---} > [].$

$a(s(N)) \text{ ---} > [a], a(N).$

$b(0) \text{ ---} > [].$

$b(s(N)) \text{ ---} > [b], b(N).$

$c(0) \text{ ---} > [].$

$c(s(N)) \text{ ---} > [c], c(N).$

?–  $abc(X, []).$

$X = [] ;$

$X = [a, b, c] ;$

$X = [a, a, b, b, c, c] ;$

## GENERATIVNÍ SÍLA DCG

Generativní (rozpoznávací) síla DCG je větší než CFG

např. jazyk  $a^n b^n c^n$ :

```
abc --> a(N), b(N), c(N).
```

```
a(0) --> [].
```

```
a(s(N)) --> [a], a(N).
```

```
b(0) --> [].
```

```
b(s(N)) --> [b], b(N).
```

```
c(0) --> [].
```

```
c(s(N)) --> [c], c(N).
```

```
?- abc(X,[]).
```

```
X = [] ;
```

```
X = [a, b, c] ;
```

```
X = [a, a, b, b, c, c] ;
```

```
X = [a, a, a, b, b, b, c, c, c] ;
```

```
...
```

---

---

✓ ●	Komunikace . . . . .	2
✓ ●	Gramatiky . . . . .	6
⇒ ●	Analýza přirozeného jazyka . . . . .	25
●	PA026 – Projekt z umělé inteligence . . . . .	33

## VÝZNAM SYNTAKTICKÉ ANALÝZY

→ analýza syntaxe je **nutná** pro analýzu **významu**

→ většina teorií analýzy významu dodržuje **princip kompozicionality**:

*Význam složeného výrazu je funkcí významu jednotlivých podvýrazů*

→ **proces** sémantické analýzy:

– buď vychází z **výsledků** syntaktické analýzy

– nebo **probíhá současně** se syntaktickou analýzou; pak může zasahovat i do tvorby syntaktického stromu

## PROBLÉMY PŘI ANALÝZE PŘIROZENÉHO JAZYKA

- víceznačnost
- anaforické výrazy
- indexické výrazy
- nejasnost
- nekompozicionalita
- struktura promluvy
- metonymie
- metafora

## VÍCEZNAČNOST

→ *ambiguity*

→ **víceznačnost** může být **lexikální**, **syntaktická**, **sémantická** a **referenční**

## VÍCEZNAČNOST

- *ambiguity*
- **víceznačnost** může být **lexikální**, **syntaktická**, **sémantická** a **referenční**
- lexikální – “stát,” “žena,” “hnát”

## VÍCEZNAČNOST

- *ambiguity*
- **víceznačnost** může být **lexikální**, **syntaktická**, **sémantická** a **referenční**
- lexikální – “stát,” “žena,” “hnát”
- syntaktická – “Jím špagety s masem.”



## VÍCEZNAČNOST

- *ambiguity*
- **víceznačnost** může být **lexikální**, **syntaktická**, **sémantická** a **referenční**
- lexikální – “stát,” “žena,” “hnát”
- syntaktická – “Jím špagety s masem.”  
“Jím špagety se salátem.”

## VÍCEZNAČNOST

- *ambiguity*
- **víceznačnost** může být **lexikální**, **syntaktická**, **sémantická** a **referenční**
- lexikální – “stát,” “žena,” “hnát”
- syntaktická – “Jím špagety s masem.”  
“Jím špagety se salátem.”  
“Jím špagety s použitím vidličky.”

## VÍCEZNAČNOST

- *ambiguity*
- **víceznačnost** může být **lexikální**, **syntaktická**, **sémantická** a **referenční**
- lexikální – “stát,” “žena,” “hnát”
- syntaktická – “Jím špagety s masem.”  
“Jím špagety se salátem.”  
“Jím špagety s použitím vidličky.”  
“Jím špagety se sebezapřením.”

## VÍCEZNAČNOST

- *ambiguity*
- **víceznačnost** může být **lexikální**, **syntaktická**, **sémantická** a **referenční**
- lexikální – “stát,” “žena,” “hnát”
- syntaktická – “Jím špagety s masem.”  
“Jím špagety se salátem.”  
“Jím špagety s použitím vidličky.”  
“Jím špagety se sebezapřením.”  
“Jím špagety s přítelem.”

## VÍCEZNAČNOST

- *ambiguity*
- **víceznačnost** může být **lexikální**, **syntaktická**, **sémantická** a **referenční**
- lexikální – “stát,” “žena,” “hnát”
- syntaktická – “Jím špagety s masem.”  
“Jím špagety se salátem.”  
“Jím špagety s použitím vidličky.”  
“Jím špagety se sebezapřením.”  
“Jím špagety s přítelem.”
- sémantická – “**Jeřáb** je vysoký.” “Viděli jsme veliké **oko**.”

## VÍCEZNAČNOST

→ *ambiguity*

→ **víceznačnost** může být **lexikální**, **syntaktická**, **sémantická** a **referenční**

→ lexikální – “stát,” “žena,” “hnát”

→ syntaktická – “Jím špagety s masem.”

“Jím špagety se salátem.”

“Jím špagety s použitím vidličky.”

“Jím špagety se sebezapřením.”

“Jím špagety s přítelem.”

→ sémantická – “**Jeřáb** je vysoký.” “Viděli jsme veliké **oko**.”

→ referenční – “**Oni** přišli pozdě.” “Můžeš mi půjčit **knihu**?” “Ředitel vyhodil dělníka, protože (**on**) byl agresivní.”

## ANAFORICKÉ A INDEXICKÉ VÝRAZY

anaforické výrazy:

→ *anaphora*

→ používají **zájmena** pro odkazování na objekty zmíněné **dříve**

## ANAFORICKÉ A INDEXICKÉ VÝRAZY

anaforické výrazy:

→ *anaphora*

→ používají **zájmena** pro odkazování na objekty zmíněné **dříve**

→ “Poté co se Honza s Marií rozhodli se vzít, (**oni**) vyhledali kněze, aby **je** oddal.”



## ANAFORICKÉ A INDEXICKÉ VÝRAZY

anaforické výrazy:

→ *anaphora*

→ používají **zájmena** pro odkazování na objekty zmíněné **dříve**

→ “Poté co se Honza s Marií rozhodli se vzít, (**oni**) vyhledali kněze, aby **je** oddal.”



## ANAFORICKÉ A INDEXICKÉ VÝRAZY

anaforické výrazy:

→ *anaphora*

→ používají **zájmena** pro odkazování na objekty zmíněné **dříve**

→ “Poté co se Honza s Marií rozhodli se vzít, (**oni**) vyhledali kněze, aby **je** oddal.”



→ “Marie uviděla ve výloze prstýnek a požádala Honzu, aby **jí ho** koupil.”

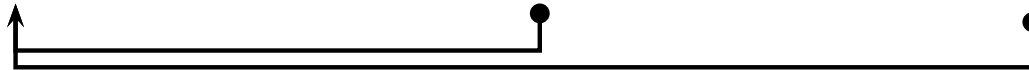
## ANAFORICKÉ A INDEXICKÉ VÝRAZY

anaforické výrazy:

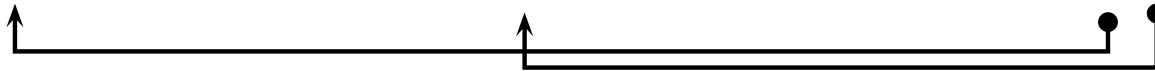
→ *anaphora*

→ používají **zájmena** pro odkazování na objekty zmíněné **dříve**

→ “Poté co se Honza s Marií rozhodli se vzít, (**oni**) vyhledali kněze, aby **je** oddal.”



→ “Marie uviděla ve výloze prstýnek a požádala Honzu, aby **jí ho** koupil.”



## ANAFORICKÉ A INDEXICKÉ VÝRAZY

### anaforické výrazy:

→ *anaphora*

→ používají **zájmena** pro odkazování na objekty zmíněné **dříve**

→ “Poté co se Honza s Marií rozhodli se vzít, (**oni**) vyhledali kněze, aby **je** oddal.”



→ “Marie uviděla ve výloze prstýnek a požádala Honzu, aby **jí ho** koupil.”



### indexické výrazy:

→ *indexicals*

→ odkazují se na údaje v **jiných částech** promluvy

## ANAFORICKÉ A INDEXICKÉ VÝRAZY

### anaforické výrazy:

→ *anaphora*

→ používají **zájmena** pro odkazování na objekty zmíněné **dříve**

→ “Poté co se Honza s Marií rozhodli se vzít, (**oni**) vyhledali kněze, aby **je** oddal.”



→ “Marie uviděla ve výloze prstýnek a požádala Honzu, aby **jí ho** koupil.”



### indexické výrazy:

→ *indexicals*

→ odkazují se na údaje v **jiných částech** promluvy

→ “**Já** jsem **tady**.”

## ANAFORICKÉ A INDEXICKÉ VÝRAZY

### anaforické výrazy:

→ *anaphora*

→ používají **zájmena** pro odkazování na objekty zmíněné **dříve**

→ “Poté co se Honza s Marií rozhodli se vzít, (**oni**) vyhledali kněze, aby **je** oddal.”



→ “Marie uviděla ve výloze prstýnek a požádala Honzu, aby **jí ho** koupil.”



### indexické výrazy:

→ *indexicals*

→ odkazují se na údaje v **jiných částech** promluvy

→ “**Já** jsem **tady**.”

→ “Proč **jsi to** udělal?”

## METAFORA A METONYMIE

metafora:

→ *metaphor*

→ použití slov v **přeneseném významu** (na základě podobnosti), často systematicky

## METAFORA A METONYMIE

metafora:

- *metaphor*
- použití slov v **přeneseném významu** (na základě podobnosti), často systematicky
- “Zkoušel jsem ten proces **zabít**, ale nešlo to.”



## METAFORA A METONYMIE

metafora:

- *metaphor*
- použití slov v **přeneseném významu** (na základě podobnosti), často systematicky
- “Zkoušel jsem ten proces **zabít**, ale nešlo to.”
- “Bouře se **vzteká**.”

## METAFORA A METONYMIE

### metafora:

- *metaphor*
- použití slov v **přeneseném významu** (na základě podobnosti), často systematicky
- “Zkoušel jsem ten proces **zabít**, ale nešlo to.”
- “Bouře se **vzteká**.”

### metonymie:

- *metonymy*
- používání **jména** jedné **věci** pro (často zkrácené) označení **věci jiné**

## METAFORA A METONYMIE

### metafora:

- *metaphor*
- použití slov v **přeneseném významu** (na základě podobnosti), často systematicky
- “Zkoušel jsem ten proces **zabít**, ale nešlo to.”
- “Bouře se **vzteká**.”

### metonymie:

- *metonymy*
- používání **jména** jedné **věci** pro (často zkrácené) označení **věci jiné**
- “Čtu **Shakespeara**.”

## METAFORA A METONYMIE

### metafora:

- *metaphor*
- použití slov v **přeneseném významu** (na základě podobnosti), často systematicky
- “Zkoušel jsem ten proces **zabít**, ale nešlo to.”
- “Bouře se **vzteká**.”

### metonymie:

- *metonymy*
- používání **jména** jedné **věci** pro (často zkrácené) označení **věci jiné**
- “Čtu **Shakespeara**.”
- “**Chrysler** oznámil rekordní zisk.”

## METAFORA A METONYMIE

### metafora:

- *metaphor*
- použití slov v **přeneseném významu** (na základě podobnosti), často systematicky
- “Zkoušel jsem ten proces **zabít**, ale nešlo to.”
- “Bouře se **vzteká**.”

### metonymie:

- *metonymy*
- používání **jména** jedné **věci** pro (často zkrácené) označení **věci jiné**
- “Čtu **Shakespeara**.”
- “**Chrysler** oznámil rekordní zisk.”
- “Ten **pstruh na másle** u stolu 3 chce další pivo.”

## NEKOMPOZICIONALITA

- *noncompositionality*
- příklady **porušení pravidla kompozicionality** u ustálených termínů nebo přednost jiného možného významu při určitých spojeních

## NEKOMPOZICIONALITA

- *noncompositionality*
- příklady **porušení pravidla kompozicionality** u ustálených termínů nebo přednost jiného možného významu při určitých spojeních
- “aligátoří boty,” “basketbalové boty,” “dětské boty”

## NEKOMPOZICIONALITA

- *noncompositionality*
- příklady **porušení pravidla kompozicionality** u ustálených termínů nebo přednost jiného možného významu při určitých spojeních
- “aligátoří boty,” “basketbalové boty,” “dětské boty”
- “pata sloupu”



## NEKOMPOZICIONALITA

- *noncompositionality*
- příklady **porušení pravidla kompozicionality** u ustálených termínů nebo přednost jiného možného významu při určitých spojeních
- “aligátoří boty,” “basketbalové boty,” “dětské boty”
- “pata sloupu”
- “červená kniha,” “červené pero”

## NEKOMPOZICIONALITA

- *noncompositionality*
- příklady **porušení pravidla kompozicionality** u ustálených termínů nebo přednost jiného možného významu při určitých spojeních
- “aligátoří boty,” “basketbalové boty,” “dětské boty”
- “pata sloupu”
- “červená kniha,” “červené pero”
- “bílý trpaslík”

## NEKOMPOZICIONALITA

- *noncompositionality*
- příklady **porušení pravidla kompozicionality** u ustálených termínů nebo přednost jiného možného významu při určitých spojeních
- “aligátoří boty,” “basketbalové boty,” “dětské boty”
- “pata sloupu”
- “červená kniha,” “červené pero”
- “bílý trpaslík”
- “dřevěný pes,” “umělá tráva”

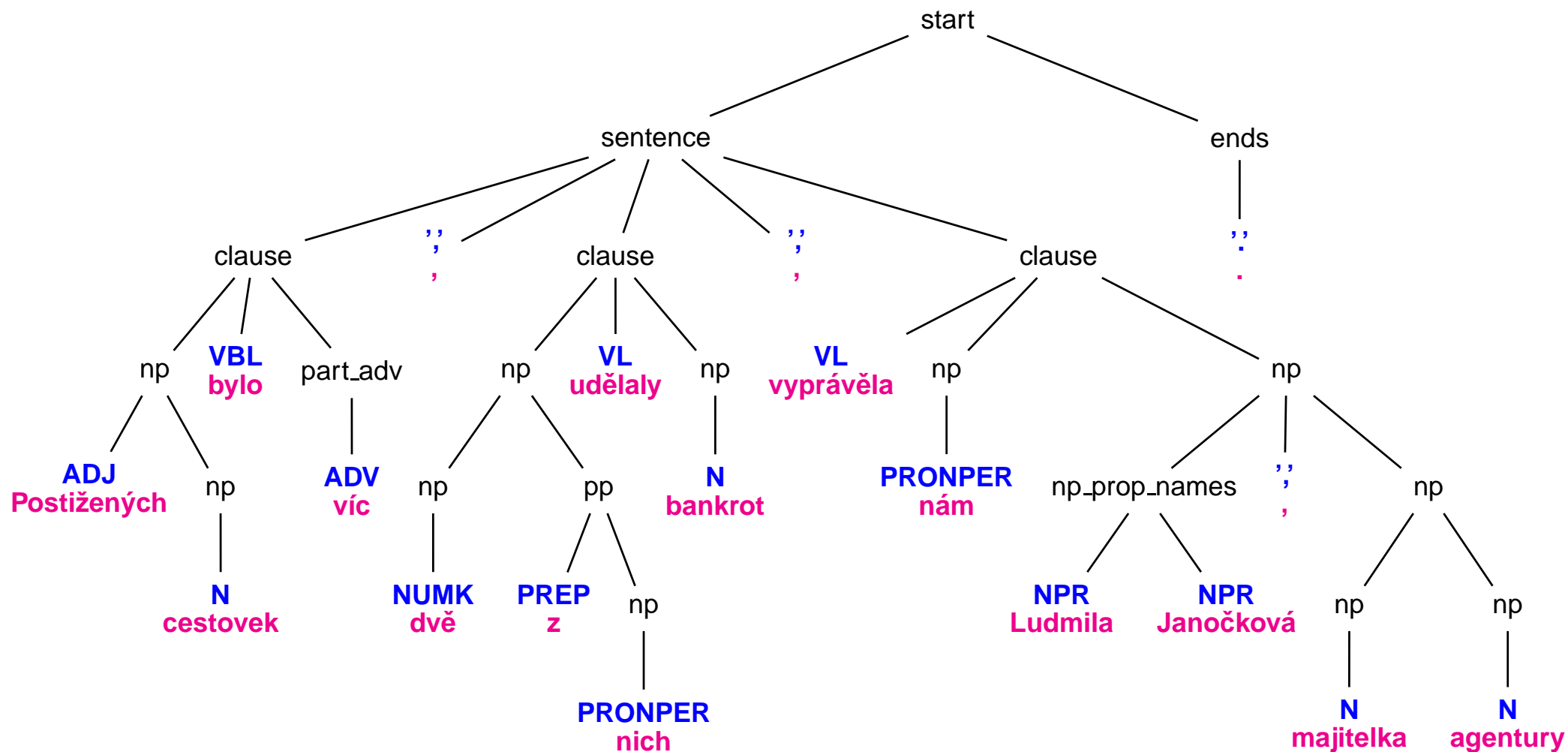
## NEKOMPOZICIONALITA

- *noncompositionality*
- příklady **porušení pravidla kompozicionality** u ustálených termínů nebo přednost jiného možného významu při určitých spojeních
- “aligátoří boty,” “basketbalové boty,” “dětské boty”
- “pata sloupu”
- “červená kniha,” “červené pero”
- “bílý trpaslík”
- “dřevěný pes,” “umělá tráva”
- “velká molekula”

## REÁLNÁ SYNTAKTICKÁ ANALÝZA PŘIROZENÉHO JAZYKA

- velice rozsáhlé gramatiky (desítky až stovky tisíc pravidel)
- silná víceznačnost – někdy až obrovské množství ( $>$  milióny) možných syntaktických stromů  
*Obehnat Šalounův pomník mistra Jana Husa na pražském Staroměstském náměstí živým plotem z hustých keřů s trny navrhuje občanské sdružení Společnost Jana Jesenia.*
- existují efektivní algoritmy pro takové gramatiky  
např. tabulkový analyzátor (*chart parser*), běží v  $O(n^3)$ , tisíce slov/sekundu

## PŘÍKLAD STROMU ANALÝZY V SYSTÉMU SYNT



<http://nlp.fi.muni.cz/projekty/wwwsynt/>

---

---

✓ ●	Komunikace . . . . .	2
✓ ●	Gramatiky . . . . .	6
✓ ●	Analýza přirozeného jazyka . . . . .	25
⇒ ●	PA026 – Projekt z umělé inteligence . . . . .	33

## PA026 – PROJEKT Z UMĚLÉ INTELIGENCE

- navazuje na předmět *PB016 Úvod do umělé inteligence*
- volba programovacího jazyka ovšem není nijak omezena
- samostatná volba tématu v rozsahu  $\geq 1$  semestru
- předmět probíhá jako konzultace
- zajímavé výsledky (<http://nlp.fi.muni.cz/projekty/...>)
  - ⇨ projekt [elnet](#) – > 5 let spolupráce na grantových projektech simulace elektrorozvodných sítí
  - ⇨ projekt [plagiaty\\_z\\_webu](#) – reálné a funkční vyhledávání shod s dokumenty na celém webu
  - ⇨ projekt [robot\\_johnny\\_5](#) – sestavení a “oživení” robota – mobilního počítače

