4 Počítačové aspekty

hardware: procesor, vnitřní paměť, periférie (disky, pásky, klávesnice, obrazovka, myš)

software: data a programy

programovací jazyky : nižší, vyšší, lingvistické formalismy = jazyky pro lingvistické specifikace

oddělení dat a algoritmů : obecný algoritmus dokáže zpracovat různá data (sčítání různých čísel, hledání v různých slovnících, analýza a syntéza různých jazyků); ideální stav: data = lingvistické specifikace, algoritmy = informatika

deklarativní a procedurální popis : dělení zhruba odpovídá rozdílu mezi daty a algoritmy, lingvistické specifikace bývají deklarativní a počítačové specifikace procedurální, avšak některá data lze interpretovat jako popis algoritmu

modularita : moduly stanoveny raději lingvisticky než programátorsky, raději žádná ad hoc řešení; jak budou modely mezi sebou komunikovat? ˙ datové struktury

robustní koncepce systému : (1) aby se při jeho vývoji nemuselo stále začínat od nuly, (2) aby se systém nezhroutil při neočekávaném vstupu

vstup a výstup : elektronicky/tisk/řeč; dávkově/interaktivně (dotaz v případě chybějící informace nebo více možností řešení); problémy s kódováním (písmo, formát)

lexikální databáze : na rozdíl od slovníků pro lidského uživatele musí být lexikální databáze pro zpracování přirozeného jazyka explicitní; jednojazyčné a překladové slovníky; komerční systémy mají běžně alespoň 15 tis. hesel obecné slovní zásoby a 15 tis. hesel terminologie; konverze z dostupných zdrojů do patřičného formátu; software pro udržbu a doplňování slovníku; snaha o "recyklovatelnost": nezávislost na konkrétní aplikaci a lingvistické teorii

Typické údaje v jednojazyčném slovníku: kmen, vzor, slovní druh, valenční rámec, sémantická omezení možných aktantů

Ukázka slovníku systému APAČ

```
ARM == N(RAMEN(NO2),*C,*PART).
ARRANGE == V(USPOR3A2D(50N),IL,*FIN,0).
ARRANGEMENT == N(USPOR3A2DA2NI2(N12),*A,*RACT,*NVT,*UNCC,*METH,0).
ARRAY == ARRANGEMENT.
ARRIVE == V(DOSPI2V(50I33,DOSPE3J),I,*FIN,*PRO,AT(K3),7,0).
ARTICLE == N(C3LA2NK(M04),G,ON(06),?(06)).
ASCERTAIN == FIND.
ASCRIBE == V(PR3IPISUJ(30N),IRI,*IO,*FIN,*PRO,0).
ASK == V(Z3A2D(50N),IJ,*FIN,*SCE,FOR(04),*PRO,0).
-- == V(PT(51N),1(*H,/),2(*SS,*A,*H,/),*PAT(2),*RFXX,*AUTH,ABOUT(NA4),AFTER(NA4),*QUOT,0).
ASPECT == N(HLEDISK(N08),*A,*METH,0).
ASSEMBLE == V(MONTUJ(30N),1(M),2(*OB,T),*FIN,*VAD2).
-- == V(SHROMA2Z3D(43P30,SHROMAZ3D3UJ),1(P),2(I,*PART,*INST),*RFX,*FIN,*VAD2).
ASSEMBLY == N(SESTAVENI2(N12),B,*NVT,0).
ASSERT == V(PROSAZUJ(30N),IR,*THT,*AUTH,0).
ASSESS == V(ODHADUJ(30I21,ODHADN),IL,*THT,AT(NA4),5,*AUTH,0).
ASSESSMENT == N(ODHAD(M03),*A,*RACT,*NVT,*UNCC).
ASSIGN == V(URC3(42N),IRI,*IO,*FIN,*PRO,0).
ASSIGNMENT == N(STANOVENI2(N12),F,*RACT,*METH,*NVT,0).
ASSIST == V(POMA2H(50N),IN,*PAT(3),*ACIF,IN(PR3I6),0).
ASSISTANT == N(ASISTENT(A22),*H,*NA,0).
-- == AD(POMOCNE2(3),M,*NA,*OGRD,0).
```

Ukázka jednojazyčného slovníku v systému ELU

```
#Word reach                  (SibStemV Verb) Frame=advl./pp-obj_/n/n.ppfor-iobj_n/n_pcle_
                             VP=[2C,3A,6A,12B,13B,15B] TI=ti Status=1 Usage=[s,s] RevDat=930118
#Word receipt                (CStemN CountUncount Noun) Status=1 Usage=[s,s] RevDat=930118
#Word receive                (EStemV Verb) Frame=n VP=[2A,6A] TI=ti Status=1 Usage=[s,s]
                             RevDat=950531
#Word reciprocity            (NoPlN Uncount Noun) Status=1 Usage=[s,s] RevDat=930118
#Word refer                  (DblStemV Verb) Frame=pp-obj_/n_pp_./n_pcle_ Prep-Obj=to
                             Pcle-Obj=back VP=[3A,14,15B] TI=ti Status=1 Usage=[s,s] RevDat=950927
#Word regard                 (CStemN CountUncount Noun) Status=1 Usage=[s,s] RevDat=930118
#Word regard                 (CStemV Verb) Frame=n/n_pp_./n_ping-objc_/n_pp-objc_
                             Prep-Objc=as VP=[6A,14,16B] TI=t Status=1 Usage=[s,s] RevDat=950426
#Word regardless             (NoInflJ Adj) Status=1 Usage=[s,s] RevDat=930118
#Word regardless_of          (Prep) Status=1 Usage=[s,s] RevDat=950620
#Word remainder              (CStemN Count Noun) Status=1 Usage=[s,s] RevDat=930118
#Word replace                (EStemV Verb) Frame=n/n_pp_./n_advl VP=[6A,14,15A] TI=t
                             Status=1 Usage=[s,s] RevDat=930118
#Word reputation             (CStemN CountUncount Noun) Status=1 Usage=[s,s] RevDat=930118
#Word request                (CStemN CountUncount Noun) Status=1 Usage=[s,s] RevDat=930118
#Word request                (CStemV Verb) Frame=n/thcls/n_inf. VP=[6A,9,17] TI=t Status=1
                             Usage=[s,s] RevDat=930118
#Word require                (EStemV Verb) Frame=n/ing/thcls/n_pp_./n_inf. VP=[6A,6D,9,14,17]
                             TI=t Status=1 Usage=[s,s] RevDat=930118
#Word responsibility         (YStemN CountUncount Noun) Status=1 Usage=[s,s] RevDat=930118
#Word restriction            (CStemN CountUncount Noun) Status=1 Usage=[s,s] RevDat=930118
#Word right                  (Adv) Status=1 Usage=[s,s] RevDat=930118
#Word right                  (CStemN CountUncount Noun) Status=1 Usage=[s,s] RevDat=930118
#Word right                  (NoInflJ Adj) Status=1 Usage=[s,s] RevDat=930118
#Word rule                   (CStemN CountUncount Noun) Status=1 Usage=[s,s] RevDat=930118
#Word rule                   (EStemV Verb)
                             Frame=i/advlmsr/advl./pp-obj_/n/thcls/n_advl/n_pcle_/n_inf-objc.
                             VP=[2A,2B,2C,3A,6A,9,15A,15B,25] TI=ti Status=1 Usage=[s,s]
                             RevDat=930118
```

Analýza (parsing) – VSTUP: gramatika, slovník, data (věta); VÝSTUP: syntaktická struktura; METODY: top-down / bottom-up, depth-first (backtracking) / breadth-first

Datové struktury : stromy, sestavy rysů pro reprezentaci vět, řetězový graf (chart) pro záznam paralelních struktur

Ukázka reprezentace věty v systému APAČ

```
-001-$sn(13)+.($gs(c,4e,s))+the+program+reads+the+data+qpiod+>-002-

-001-$($sn(13),$gs(c,4e,s))+
      s(v(c3t(sl(15i15,#pr3e),/,*sg),
         n(program(l,$ag,*def,*a,jm(m01),/,*sg)),
         n(dat(r,$pat,*def,*sp,*a,jm(n15),/,*pl))
      ))
+.-002-

-001-$sn(13)+$gs(c,4e,s)+program+c3te+data+.-002-
```

Ukázka systému ELU

Slovník

```
#Declare       Form = <head form>  Category = cat
#Restrict Cat  <* cat> = Cat
#Lookup Czech: <* cat> = verb/noun/prep

#Lexicon Czech

nemoc   * noun
      <* head lex> = nemoc
      <* head num> = sg
      <* head case> = nom/acc

probíhá * verb
      <* head lex> = probíhat
      <* head subj cat> = noun
      <* head subj head num> = sg
      <* head subj head case> = nom
      <* head obj cat> = noun
      <* head obj head case> = acc
      <* head advb cat> = pp

bez     * prep
      <* head lex> = bez
      <* head obj cat> = noun
      <* head obj head case> = gen

bez     * noun
      <* head lex> = bez
      <* head num> = sg
      <* head case> = nom/acc

příznaků * noun
      <* head lex> = příznak
      <* head num> = pl
      <* head case> = gen
```

Gramatika

```
#Grammar

#Rule S
S -> X H
   <S cat> = s
   <S head> = <H head>
   <H cat> = vp/verb
   <H head subj/obj/advb> = X
   <X topic> = yes

#Rule VP
VP -> H X
   <VP cat> = vp
   <VP head> = <H head>
   <H cat> = verb
   <H head subj/obj/advb> = X
   <X topic> = no
```

```
#Rule N
N -> H X
   <N cat> = noun
   <N head> = <H head>
   <H cat> = noun
   <H head attr> = X
   <X cat> = noun
   <X head case> = gen

#Rule PP
PP -> P H
   <PP cat> = pp
   <PP pform> = <P head lex>
   <PP head> = <H head>
   <P cat> = prep
   <P head obj> = H
```

Hledání ve slovníku

```
Looking up 'nemoc'
-------------------------------------------------
nemoc

<1>
cat = noun
head : case = acc/nom
       form = nemoc
       lex = nemoc
       num = sg

'nemoc' has 1 representation(s) in Czech
--------------------------------------------------

Looking up 'probíhá'
--------------------------------------------------
probíhá

<1>
cat = verb
head : advb : cat = pp
       form = probíhá
       lex = probíhat
       obj : cat = noun
             head : case = acc
       subj : cat = noun
              head : case = nom
                     num = sg

'probíhá' has 1 representation(s) in Czech
---------------------------------------------------
```

```
Looking up 'bez'
---------------------------------------------------
bez

<1>
cat = noun
head : case = acc/nom
       form = bez
       lex = bez
       num = sg

bez

<1>
cat = prep
head : form = bez
       lex = bez
       obj : cat = noun
             head : case = gen

'bez' has 2 representation(s) in Czech
------------------------------------------------

Looking up 'příznaků'
------------------------------------------------
příznaků

<1>
cat = noun
head : case = gen
       form = příznaků
       lex = příznak
       num = pl

'příznaků' has 1 representation(s) in Czech

--------------------------------------------------
```

Analýza věty

```
(0) nemoc (1) probíhá (2) bez (3) příznaků (4)

NUMBER OF PARSES: 4  (2 and 4, 1 and 3 unify)
Parse # 1

   ____S_____
  /          \
nemoc     ___VP___
         /        \
      probíhá   __N__
               /     \
              bez příznaků

<21>
cat = s
head : advb : cat = pp
       form = probíhá
       lex = probíhat
       obj : cat = noun
```

```
            head : case = acc
                   form = nemoc
                   lex = nemoc
                   num = sg
            topic = yes
      subj : cat = noun
             head : attr : cat = noun
                           head : case = gen
                                  form = příznaků
                                  lex = příznak
                                  num = pl
                    case = nom
                    form = bez
                    lex = bez
                    num = sg
             topic = no

Parse # 2

   ____S_____
  /          \
nemoc     ___VP___
        /        \
     probíhá   __N__
              /     \
             bez příznaků

<21>
cat = s
head : advb : cat = pp
      form = probíhá
      lex = probíhat
      obj : cat = noun
            head : attr : cat = noun
                          head : case = gen
                                 form = příznaků
                                 lex = příznak
                                 num = pl
                   case = acc
                   form = bez
                   lex = bez
                   num = sg
            topic = no
      subj : cat = noun
             head : case = nom
                    form = nemoc
                    lex = nemoc
                    num = sg
             topic = yes

Parse # 3

   ____S_____
  /          \
nemoc     ___VP___
        /        \
     probíhá   _PP__
              /     \
             bez příznaků
```

```
<21>
cat = s
head : advb : cat = pp
              head : case = gen
                     form = příznaků
                     lex = příznak
                     num = pl
              pform = bez
              topic = no
       form = probíhá
       lex = probíhat
       obj : cat = noun
             head : case = acc
                    form = nemoc
                    lex = nemoc
                    num = sg
             topic = yes
       subj : cat = noun
              head : case = nom
                     num = sg

Parse # 4

   ____S_____
  /          \
nemoc     ___VP___
         /        \
      probíhá   _PP__
               /     \
              bez příznaků

<21>
cat = s
head : advb : cat = pp
              head : case = gen
                     form = příznaků
                     lex = příznak
                     num = pl
              pform = bez
              topic = no
       form = probíhá
       lex = probíhat
       obj : cat = noun
             head : case = acc
       subj : cat = noun
              head : case = nom
                     form = nemoc
                     lex = nemoc
                     num = sg
              topic = yes
```