Skannerite elust

Vesal Vojdani
(TÜ Arvutiteaduse Instituut)

- Regulaaravaldised üle (lõpliku) tähestiku Σ

$$E ::= \varepsilon \mid \mathrm{a} \mid (E\ E) \mid (E \mid E) \mid E^\star$$

kus $\mathrm{a} \in \Sigma$.

- Regulaaravaldis E defineerib keele $L(E) \subseteq \Sigma^\star$

$$s \in L(E)$$

Regulaaravaldis defineerib keele, aga leksimine…

Leksimine!

- x+++++y
  <ID:x>, <Op:++>, <Op:++>, <Op:+>, <ID:y>

Probleemiks on siin “tüübiviga”. (lvalue/rvalue)

- x+++ ++y
  <ID:x>, <Op:++>, <Op:+>, <Op:++>, <ID:y>

See töötaks, aga lekser teeb oma töö ja **teised enam ei vaata tagasi**!

Need ei vaata tagasi:

- Talulaps
- Oja süda
- Java kompilaator
- ANTLR

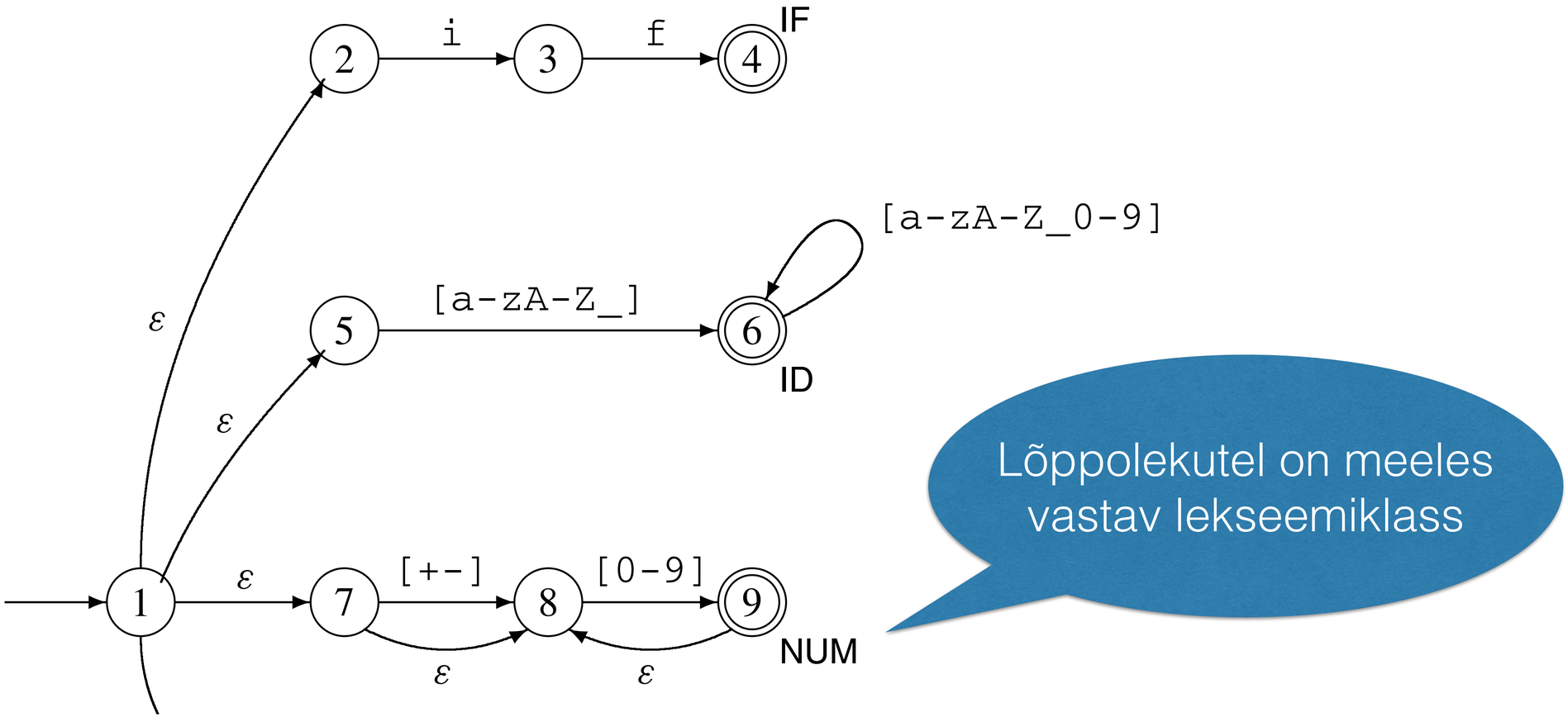
Kõige rohkem probleeme on meil aga ANTLRiga!
Kui lekseri töö on tehtud, siis enam token stream'ile muudatusi ei tehta!

Analoogia…

- https://www.newyorker.com/culture/culture-desk/a-few-words-about-that-ten-million-dollar-serial-comma
- “The canning, processing, preserving, freezing, drying, marketing, storing, packing for shipment or distribution of [milk need not be paid]”
- Kas nendele, kes tegelevad ainult "distribution"-ga, võib jätta ületundite eest maksmata?
- Longest munch: "packing for shipment or distribution"
- Või kaks tükki: "packing for shipment", "distribution".

Leksiline spetsifikatsioon

- Keyword: ‘if’ | …
- Op: ‘++’ | ‘+’ | …
- Identifier: [a-zA-Z_][a-zA-Z_0-9]*
- …



Kombineeritud keel

$E = E_1 \mid E_2 \mid \ldots E_n$

Kuidas kasutada

- Sisendiks on $x_1 \ldots x_m$.
- Otsime sellist *i*, et alamsõne $w = x_1 \ldots x_i$ kuuluks keelde L(E).
- Kui leidub, siis peab olema alamkeel $L(E_j)$, kuhu sõne kuulub. (Automaadi lõppolekus oli kirjas: E oli ju $E_1 \mid E_2 \mid \ldots E_n$.)
- Eemaldame sisendist sõne w ja jätkame kuni sisend on tühi.

Maximal Munch!

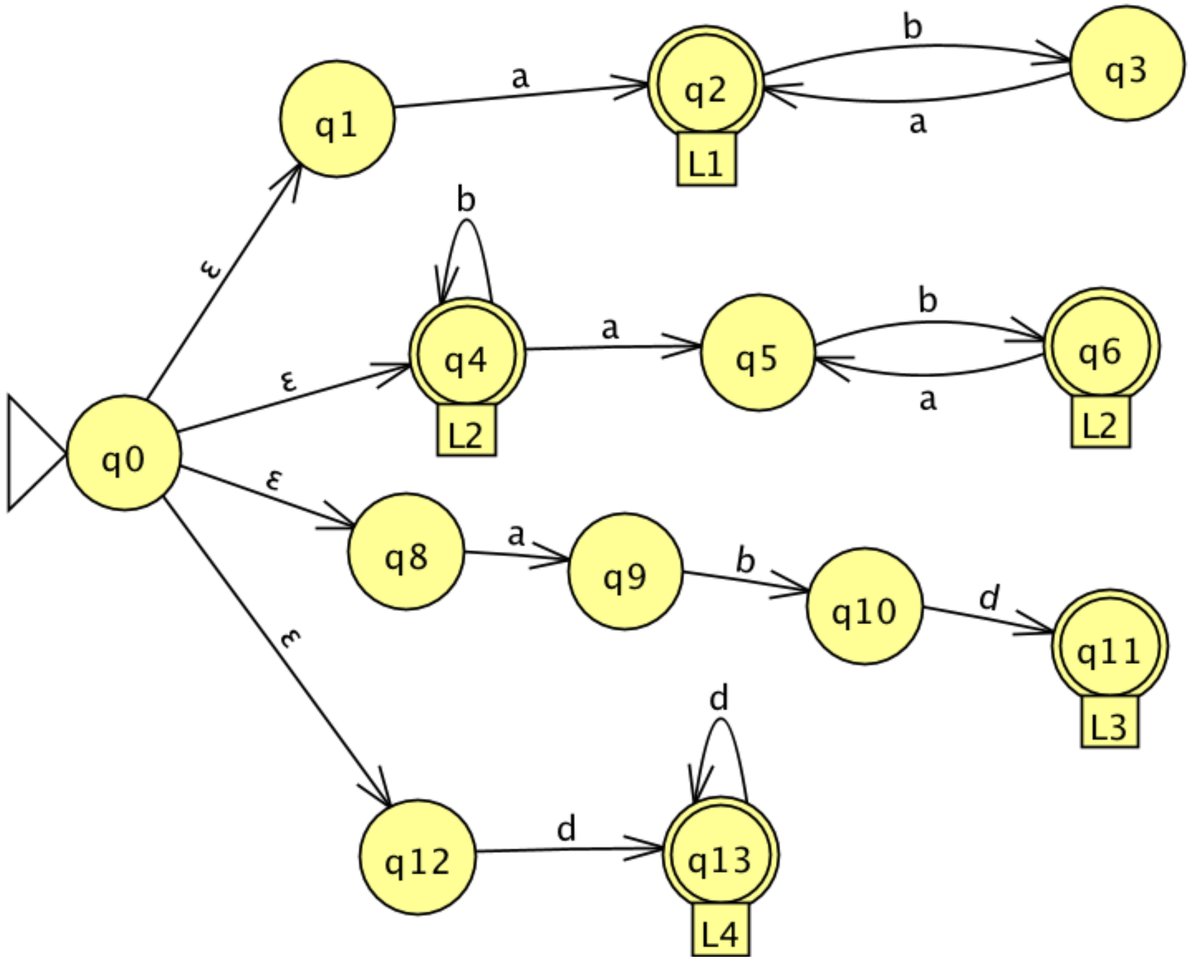
- Mis juhtub, kui sobivad kaks alamsõne??
- Võib ju juhtuda, et $x_1 \ldots x_i$ ja $x_1 \ldots x_j$ mõlemad kuuluvad keelde L(E).
- Näiteks ‘++’ ja ‘+’ on mõlemad Java operaatorid.
- Loomulik konventsioon on valida pikim alamsõne, mis sobitub!

Milline lekseemiklass?

- Kas “`if`” on muutuja nimi või keyword?
- Prioriteedijärjekord: leksilise spetsifikatsiooni järjekord on oluline.
- NB! Kõigepealt valime pikim alamsõne ja alles siis valime lekseemiklass. (Seega, “`ifo`” on muutuja.)

Quiz

```
L1: 'a' 'ba'*;
L2: 'b'* 'ab'*;
L3: 'abd';
L4: 'd'+;
```



Käsitsi tehes (kodutöö)

```
StringBuilder sb = new StringBuilder();
while (Character.isLetter(peek())) {
    sb.append(peek());
    consume();
}
String identOrIf = sb.toString();

if (identOrIf.equals("if")) {
    return new Token(IF);
} else {
    return new Token(IDENT, identOrIf);
}
```

Siin loeme kuni "automaat" sureb.

Tähtis on siin aru saada, mida lekser peab tegema (**maximal munch**), aga selleks me ei pea päris lekserit simuleerima.