Linguagens Formais e Autômatos

Aula 16 - Linguagens não livres de contexto

Prof. Dr. Daniel Lucrédio
Departamento de Computação / UFSCar
Última revisão: ago/2015

Referências bibliográficas

- **Introdução à teoria dos autômatos, linguagens e computação / John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman** ; tradução da 2.ed. original de Vandenberg D. de Souza. - Rio de Janeiro : Elsevier, 2002 (Tradução de: Introduction to automata theory, languages, and computation - ISBN 85-352-1072-5)
  - Capítulo 7 - Seção 7.2
- **Introdução à teoria da computação / Michael Sipser** ; tradução técnica Ruy José Guerra Barretto de Queiroz ; revisão técnica Newton José Vieira. -- São Paulo : Thomson Learning, 2007 (Título original : Introduction to the theory of computation. "Tradução da segunda edição norte-americana" - ISBN 978-85-221-0499-4)
  - Capítulo 2 - Seção 2.3

Lema do bombeamento

- Para linguagens regulares
    - Prova que linguagens não são regulares
    - Base: para linguagens regulares, temos autômatos finitos
        - Autômatos finitos não conseguem contar
        - Esse é o limite das linguagens regulares, e basicamente é o que o lema busca provar
- Para linguagens livres de contexto
    - Prova que linguagens não são livres de contexto
    - Base: para linguagens livres de contexto, temos autômatos de pilha (veremos a seguir)
        - Autômatos de pilha conseguem contar somente uma “coisa”, mas não duas, ao mesmo tempo
        - Esse é o limite das linguagens regulares, e basicamente é o que o lema busca provar

Lema do bombeamento

- Parecido com o lema do bombeamento para linguagens regulares
  - Em LR
    - Dividimos uma cadeia s em 3 partes, xyz
    - E “bombeamos” y (isto é, fazemos xy^iz para $i \geq 0$), e a cadeia resultante ainda deve estar na linguagem
  - Para LLC
    - Dividimos uma cadeia s em 5 partes, uvwxy
    - E “bombeamos” v e x (isto é, fazemos uv^iwx^iy para $i \geq 0$), e a cadeia resultante ainda deve estar na linguagem
- Ex: Se abcdefg faz parte da linguagem
  - Fazemos s=uvwxy, u=a,v=bc,w=de,x=f,y=g
  - Então “bombeando” v e x zero ou mais vezes, sempre obtemos cadeias que fazem parte da linguagem
  - Ou seja, adeg (i=0), abcbcdeffg (i=2) e abcbcbcdefffg (i=3) fazem parte da linguagem
    - Assim como todas as outras cadeias com v e x “bombeadas”

Lema do bombeamento para linguagens livres de contexto

- Seja L uma linguagem livre de contexto.
  - Então, existe uma constante n tal que, se z é qualquer cadeia em L tal que |z| é pelo menos n, podemos escrever z = uvwxy, sujeito às seguintes condições:
    - $|vwx| \leq n$. Ou seja, a porção intermediária não é muito longa.
    - $vx \neq \varepsilon$. Tendo em vista que v e x são os fragmentos a serem "bombeados", essa condição diz que pelo menos uma das cadeias que bombeamos não deve ser vazia.
    - Para todo $i \geq 0$, uv^iwx^iy está em L. Isto é, as duas cadeias v e x podem ser "bombeadas" qualquer número de vezes, incluindo 0, e a cadeia resultante ainda será um elemento de L.

Linguagens não livres de contexto

- Linguagens que precisam "contar duas coisas"
  - Ex: correspondência entre três grupos de símbolos de igualdade
    - $\{0^n1^n2^n \mid n \geq 1\}$ (ou $0^+1^+2^+$ com um número igual de cada símbolo)
    - Exs: 012, 001122, 000111222
  - Ex: comparar dois pares com números iguais de símbolos, quando os pares se intercalam
    - $\{0^i1^j2^i3^j \mid i \geq 1 \text{ e } j \geq 1\}$
    - Exs: 00012223, 0011111223333

Linguagens não livres de contexto

- Exemplo importante
  - Linguagens livres de contexto não podem comparar duas cadeias de comprimento arbitrário, se as cadeias forem escolhidas a partir de um alfabeto com mais de um símbolo.
  - Seja L = {ww | w está em {0,1}*}. Isto é, L consiste em cadeias repetitivas, como ε, 0101, 00100010, 110110.
  - Usando o lema do bombeamento, é possível provar que L não é livre de contexto

Aplicando o lema do bombeamento para linguagens livres de contexto

- Se L é livre de contexto, então seja n a constante de seu lema do bombeamento
  - Considere a cadeia $z = 0^n 1^n 0^n 1^n$
  - Essa cadeia é $0^n 1^n$ repetida, e assim, z está em L
- Vamos desmembrar z = uvwxy, tal que $|vwx| \leq n$ e $vx \neq \varepsilon$
  - É possível mostrar que uwy não está em L
    - Não faremos essa prova aqui!
  - Ou seja, “bombeando” v e x zero vezes, obtemos uma cadeia que não está em L
    - Ferindo o lema
  - Isso é uma contradição, e portanto concluímos que L não é livre de contexto

Linguagens não livres de contexto

- Esse caso é particularmente interessante
- Considere a seguinte cadeia, em uma linguagem de programação típica:

```
String numero = 0;
if (nmero > 0) {
  System.out.println("Nunca vai entrar aqui");
}
```

Irá acusar erro aqui, pois a variável nmero não foi declarada

- Em algumas LP, variáveis precisam ser declaradas antes de serem utilizadas
  - É o mesmo caso da linguagem {ww | w em um alfabeto com mais de um símbolo}

Linguagens não livres de contexto

- Outros exemplos: declaração de pacotes, macros, chamada de funções, etc.
- Ou seja, gramáticas livres de contexto não conseguem impor todas as restrições “semânticas” de uma linguagem de programação típica
- Como é feito então?
  - Outros mecanismos, como uma “tabela de símbolos”
  - Mais sobre isso nas disciplinas de compiladores

Fim

Aula 16 - Linguagens não livres de contexto