E6 10

Universidade de Aveiro

Departamento de Eletrónica, Telecomunicações e Informática

Compiladores

Exame teórico-prático, parte 2 (Ano Letivo de 2022-2023) **Época de recurso, 22/06/2023**

Curso: [redacted]

ALGORITMO do `predict`:

$$\texttt{predict}(A \to \alpha) = \begin{cases} \texttt{first}(\alpha) & \text{se } \varepsilon \notin \texttt{first}(\alpha) \\ (\texttt{first}(\alpha) - \{\varepsilon\}) \cup \texttt{follow}(A) & \text{se } \varepsilon \in \texttt{first}(\alpha) \end{cases}$$

ALGORITMO do `first`:

```
first(α) {
   if (α = ε) then
      return {ε}
   h = head (α)        # com |h| = 1
   ω = tail (α)        # tal que α = hω
   if (h ∈ T) then
      return {h}
   else
      return    ∪           first(β_i ω)
             (h→β_i)∈P
}
```

ALGORITMO do `follow`:

1. $\$ \in \texttt{follow}(S)$
2. $\texttt{if } (A \to \alpha B \in P) \texttt{ then}$
   $\texttt{follow}(B) \supseteq \texttt{follow}(A)$
3. $\texttt{if } (A \to \alpha B \beta \in P) \wedge (\varepsilon \notin \texttt{first}(\beta)) \texttt{ then}$
   $\texttt{follow}(B) \supseteq \texttt{first}(\beta)$
4. $\texttt{if } (A \to \alpha B \beta \in P) \wedge (\varepsilon \in \texttt{first}(\beta)) \texttt{ then}$
   $\texttt{follow}(B) \supseteq ((\texttt{first}(\beta) - \{\varepsilon\}) \cup \texttt{follow}(A))$

---

1. Sobre o alfabeto $T_1 = \{\texttt{a}, \texttt{b}, \texttt{c}, \texttt{d}, \texttt{e}, \texttt{f}\}$ considere a gramática G_1 dada a seguir e seja L_1 a linguagem por ela descrita.

```
S → A B A
A → ε | a A
B → b C f | b B f | C
C → d D | c D e
D → D d | E d | ε
E → e E | f E
```

90 [1,5] (a) Faça a derivação à esquerda da palavra `a b c d e f`. Tem de apresentar todos os passos (derivações diretas).

60 [2,0] (b) Determine o conjunto `predict`(S → A B A).
Apresente os passos intermédios e/ou o raciocínio adequados para suportar a sua resposta. A simples apresentação da resposta final **não** será cotada.

60 [2,0] (c) Determine a veracidade da afirmação $\{\texttt{a}, \texttt{f}\} \subset \texttt{follow}(\texttt{B})$.
Apresente os passos intermédios e/ou o raciocínio adequados para suportar a sua resposta. A simples apresentação da resposta final **não** será cotada.

100 [2,0] (d) Um símbolo não terminal é útil se for simultaneamente acessível e produtivo; é inútil caso contrário. Analisando e apresentando a acessibilidade e a produtividade de D e de E, mostre que, na gramatica G_1, D é útil e E é inútil. Apresente os passos intermédios e/ou o raciocínio adequados para suportar a sua resposta.

65 [2,0] (e) A gramática G_1 é inadequada à implementação de um reconhecedor descendente com *lookahead* de 1. Diga porquê e altere-a de forma a obter uma equivalente que o permita. Basta transcrever as partes alteradas.

---

2. Sobre o alfabeto $T_2 = \{\texttt{a}, \texttt{b}, \texttt{c}, \texttt{d}, \texttt{e}\}$ considere a linguagem L_2 tal que:

$$L_2 = \{\ \texttt{a}^n \texttt{b}^n (\texttt{c}|\texttt{d})^{k-1} (\texttt{ee})^{k+1} \ :\ n > 0 \wedge k > 0\ \}$$

100 [1,0] (a) Apresente as 2 palavras mais pequenas (em número de letras) que pertencem a L_2.

40 [2,0] (b) Construa uma gramática livre de contexto que represente a linguagem L_2.

---

continua na página seguinte

3. Sobre o alfabeto $T_3 = \{$ `e ; < > [ ]` $\}$, considere a gramática G_3 dada a seguir, cujo símbolo inicial é o G, e seja L_3 a linguagem por ela descrita. Considere ainda os conjuntos `follow` dos seus símbolos não terminais.

R1	G	→	N
R2	N	→	e
R3		|	e < L >
R4	L	→	B
R5		|	L ; B
R6	B	→	[e] N
R7		|	N

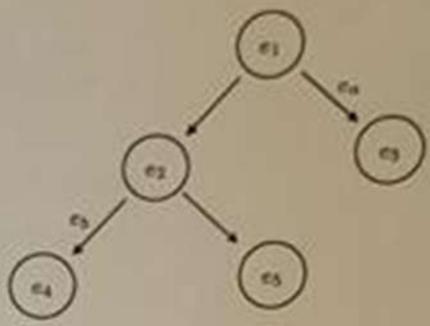
`follow(G) = { $ }`

`follow(N) = { > ; $ }`

`follow(L) = { > ; }`

`follow(B) = { > ; }`

A linguagem L_3 representa grafos acíclicos (árvores) com etiquetas nos nós e com etiquetas opcionais nas transições. Por exemplo



é uma árvore com 5 nós e 4 transições, das quais 2 não têm etiquetas. Está árvore é descrita pela palavra

`e< e< [e]e;e>;[e]e>`

ou

e_1`<`e_2`< [`e_b`]`e_4`;`e_5`>;[`e_a`]`e_3`>`

onde os índices nos símbolos terminais e foram acrescentados para tornar clara a correspondência entre a árvore e a palavra.

Considere ainda o conjunto de estados (conjuntos de itens) usado na contrução de um reconhecedor ascendente parcialmente apresentada a seguir, onde $\delta(Z_i, a)$ representa a função de transição de estado. Note que aqui, contrariamente ao que acontece no exame modelo, se optou por colocar o marcador de fim de entrada (`$`).

$$Z_0 = \{\, G \to \cdot N \,\$, \quad N \to \cdot \mathtt{e}, \quad N \to \cdot \mathtt{e} \mathtt{<} L \mathtt{>} \,\}$$
$$\delta(Z_0, N) = Z_1 = \{\, G \to N \cdot \$ \,\}$$
$$\delta(Z_0, \mathtt{e}) = Z_2 = \{\, N \to \mathtt{e} \cdot, \quad N \to \mathtt{e} \cdot \mathtt{<} L \mathtt{>} \,\}$$
$$\delta(Z_2, \mathtt{<}) = Z_3 = \{\, N \to \mathtt{e}\mathtt{<} \cdot L \mathtt{>}, \quad L \to \cdot L \,\mathtt{;}\, B, \quad L \to \cdot B, \quad B \to \cdot \mathtt{[e]}, \quad B \to \cdot N,$$
$$N \to \cdot \mathtt{e}, \quad N \to \cdot \mathtt{e} \mathtt{<} L \mathtt{>} \,\}$$
$$\delta(Z_3, L) = Z_4 = \{\, N \to \mathtt{e}\mathtt{<} L \cdot \mathtt{>}, \quad L \to L \cdot \mathtt{;}\, B \,\}$$
$$\delta(Z_3, N) = Z_5 = \{\, B \to N \cdot \,\}$$
$$\delta(Z_3, B) = Z_6 = \{ \cdots \}$$
$$\delta(Z_3, \mathtt{e}) = Z_2$$
$$\delta(Z_3, \mathtt{[}) = Z_7 = \{ \cdots \}$$
$$\delta(Z_4, \mathtt{;}) = Z_8 = \{ \cdots \}$$
$$\delta(Z_4, \mathtt{>}) = Z_9 = \{ \cdots \}$$

100 [1,5] (a) Trace a árvore de derivação da palavra `e < e < [ e ] e ; e > ; [ e ] e >`, que representa o grafo da figura.

100 [2,0] (b) Determine os conjuntos de itens definidores dos estados Z_6 a Z_9.

70

S = shift
R = reduce

[2,0] (c) Preencha as linhas da tabela de reconhecimento (*parsing*) para um reconhecedor ascendente relativamente aos estados Z_0 a Z_5. O preenchimento não depende da resposta à alínea anterior.

	e	<	>	[	]	;	$	N	L	B
Z_0	S,e									
Z_1								Z_1		
Z_2		S,<	R_2				accept R_2			
Z_3	S,e			S,[				Z_3	Z_4	Z_6
Z_4			S,>			S,;				
Z_5			R_7			R_7	~~R_7~~			

10
20
20
20

∅

[2,0] (d) Sem alterar a gramática e sem introduzir variáveis globais, construa uma gramática de atributos que associe ao símbolo inicial G 2 atributos, um que indique o número total de transições e outro o número total de transições com etiqueta que a árvore contém. No exemplo da figura, há 4 transições, 2 com etiquetas. Introduza os atributos auxiliares que necessite. Responda na tabela abaixo.

Produção	Regras semânticas
$G \to N$	~~$G.x = e_x$ ($\alpha = a$) G … $x = 0$; $G.x = N.x$; $N.x = e.x$~~
$N \to$ e	~~$N = null$~~
$N \to$ e<L>	~~$N(\alpha, x) = e_x < e_\alpha >$~~
$L \to B$	~~$L.x = B_\alpha$~~
$L_0 \to L_1$; B	~~$L.x = L_{x+1}$; B_α~~
$B \to N$	~~$B.\alpha = N.x$~~
$B \to$ [e] N	~~$B_\alpha = [e_\alpha] N.x$~~