LFTC – **Seminar 9**

1. **FIRST1, FOLLOW1**
2. Determinați FIRST1 and FOLLOW1

pentru neterminalele gramaticii următoare: S → abA

S → ε

A → Saa

A → b

...

**2. Analiza sintactică descendentă : LL(1)**

1. Dându-se gramatica:

S → if c then S endif

S → if c then S else S endif

S → stmt

Dacă înlocuim**: *if c then*** cu ***a***, **else** cu ***b***,

***endif*** cu ***c***, și ***stmt*** cu ***i*** avem:

S → a S c

S → a S b S c

S → i

Pentru una dintre cele 2 gramatici de mai sus:

a) Verificați dacă gramatica este LL(1).

b) Încercați să transformați gramatica în una echivalentă LL(1) aplicând factorizarea la stânga. Verificați dacă noua gramatică este LL(1).

c) Folosind un analizor descendent verificați dacă secvența: if c then if c then stmt else stmt endif endif (sau echivalenta ei scrisa cu a,b,c,i) aparține limbajului generat de gramatică.

a a i b i c c

a)

S → a S c (1)

S → a S b S c (2)

S → i (3)

|  |  |  |
| --- | --- | --- |
|  | First1 | Follow1 |
| S | a,i | $,c,b |

**Tabelul de analiză LL(1)**

Coloane: terminale & $

Linii: Neterminale, terminale & $...

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **a** | **i** | **c** | **b** | **$** |
| **S** | (aSc,1),(aSbSc,2) |  |  |  |  |
| **a** |  |  |  |  |  |
| **i** |  |  |  |  | S → a S c (1)  S → a S b S c (2)  S → i (3) |
| **c** |  |  |  |  |  |
| **b** |  |  |  |  |  |
| **$** |  |  |  |  |  |

Avem conflict => Gramatica nu e de tip LL(1)

b) Factorizarea la stânga:

S → a S X (1)

X → c (2)

X → b S c (3)

S → i (4)

|  |  |  |
| --- | --- | --- |
|  | First1 | Follow1 |
| S | a,i | $, c, b |
| X | c,b | $, c, b |

**Tabelul de analiză LL(1)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **a** | **b** | **c** | **i** | **$** |
| **S** | (aSX,1) | err | err | (i,4) | err  S → a S X (1)  X → c (2)  X → b S c (3)  S → i (4) |
| **X** | err | (bSc,3) | (c,2) | err | err |
| **a** | pop | err | err | err | err |
| **b** | err | pop | err | err | err |
| **c** | err | err | pop | err | err |
| **i** | err | err | err | pop | err |
| **$** | err | err | err | err | acc |

Nu sunt conflicte => Gramatica e de tip LL(1)

c) a a i b i c c ∈ L(G)?

Analizorul LL(1):

(**a**aibicc$,**S**$,ε) |– (push 1) (aaibicc$,aSX$,1) |– (pop) (aibicc$,SX$,1) |– (push 1) (aibicc$,aSXX$,11) |– (pop) (ibicc$,SXX$,11) |– (push4) (ibicc$,iXX$,114) |– (pop) (bicc$,XX$,114) |– (push 3) (bicc$,bScX$,1143) |– (pop) (icc$,ScX$,1143) |– (push 4) (icc$,icX$,11434) |– (pop) (cc$,cX$,11434) |– (pop) (c$,X$,11434) |– (push 2) (c$,c$,114342) |– (pop) ($,$,114342) |– (acc) acc => aaibicc ∈ L(G), și șirul producțiilor utilizate pentru a

genera aaibicc este: 1, 1, 4, 3, 4, 2.

1. Fie gramatica ambiguă:

S → if c then S else S

S → if c then S

S → stmt

Dacă înlocuim**: *if c then*** cu ***a***, ***else*** cu ***b***,

***endif*** cu ***c***, și ***stmt*** cu ***i*** avem:

S → a S b S

S → a S

S → i

Pentru una dintre cele 2 gramatici de mai sus:

a) Verificați dacă gramatica este LL(1).

b) Încercați să transformați gramatica în una echivalentă LL(1) aplicând factorizarea la stânga. Verificați dacă noua gramatică este LL(1).

c) Discutați, împreună cu cadrul didactic, cum se poate modifica tabelul de analiză astfel încât să se elimine conflictele.

d) Folosind analizorul LL(1) dacă secvența: if c then if c then stmt else stmt (sau echivalenta ei scrisa cu a,b,c,i) aparține limbajului generat de gramatică.

a)

S → a S b S (1)

S → a S (2)

S → i (3)

|  |  |  |
| --- | --- | --- |
|  | First1 | Follow1 |
| S | a, i | $, b |

Tabelul de analiză LL(1)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **a** | **i** | **c** | **b** | **$** |
| **S** | (aSbS,1),(aS,2) |  |  |  |  |
| **a** |  |  |  |  |  |
| **i** |  |  |  |  |  |
| **c** |  |  |  |  |  |
| **b** |  |  |  |  |  |
| **$** |  |  |  |  |  |

Avem conflict => Gramatica nu e de tip LL(1)

b)

Factorizăm la stâga:

S → a S X (1)

X → b S (2)

X → ε (3)

S → i (4)

|  |  |  |
| --- | --- | --- |
|  | First1 | Follow1 |
| S | a, i | $, b |
| X | ε, b | $, b |

Tabelul de analiză LL(1)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **a** | **b** | **i** | **$** |
| **S** | (aSX,1) |  |  |  |
| **X** |  | (bS,2), (ε,3) |  | (ε,3) |
| **a** |  |  |  |  |
| **b** |  |  |  |  |
| **i** |  |  |  |  |
| **$** |  |  |  |  |

Avem conflict => Gramatica nu e de tip LL(1)

c) Dicutăm

Tabelul de analiză LL(1)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **a** | **b** | **i** | **$** |
| **S** | (aSX,1) |  | (i,4) |  |
| **X** |  | (bS,2), ~~(ε,3)~~ |  | (ε,3) |
| **a** | pop |  |  |  |
| **b** |  | pop |  |  |
| **i** |  |  | pop |  |
| **$** |  |  |  | acc |

S-a eliminat conflictul, Gr. e de tip LL(1) și putem să o folosim la analiză.

...

1. Dându-se gramatica:

E → T + E | T

T → T \* F | F

F → (E) | a

a) Verificați dacă gramatica este LL(1).

b) Încercați să transformați gramatica în una echivalentă LL(1) aplicând factorizarea la stânga. Verificați dacă noua gramatică este LL(1).

c) Folosind un analizor descendent verificați dacă secvența: a+a aparține limbajului generat de gramatică.

...

E → T + E

E → T

T → T \* F

T → F

F → (E)

F → a

|  |  |  |
| --- | --- | --- |
|  | First1 | Follow1 |
| E | (, a | $, ) |
| T | (, a | \*, +, $, ) |
| F | (, a | \*, +, $, ) |

1. Fie gramatica:

List → id

List → id sep List

a) Verificați dacă gramatica este LL(1).

b) Încercați să transformați gramatica în una echivalentă LL(1) aplicând factorizarea la stânga. Verificați dacă noua gramatică este LL(1).

1. Fie gramatica:

S → begin SList end

S → stmt

SList → S

SList → S ; SList

a) Verificați dacă gramatica este LL(1).

b) Încercați să transformați gramatica în una echivalentă LL(1) aplicând factorizarea la stânga. Verificați dacă noua gramatică este LL(1).

...