First & Follow

Smene gramatike

(1) *RedoLoop* → **loop (** *Expression* **) {** *Statement* **redo (** *Expression* **) ;** *Statement* **}**

(2) *Expression* → *Expression* **||** *AndExpression* | *AndExpression*

(3) *AndExpression* → *AndExpression* **&&** *Term* | *Term*

(4) *Term* → **ID** | **CONST**

(5) *Statement* → *RedoLoop* | **ID =** *Expression* **;**

Uklanjanje levo-rekurzivnih smena

Generalni postupak za primer X → Xα | β :

X → βX'
X' → αX' | ε

(2) *Expression* → *Expression* || *AndExpression* | *AndExpression*

Se prevodi u:

(2) *Expression* → *AndExpression Expression'*

(2.1) *Expression'* → **||** *AndExpression Expression'* | **ε**

(3) *AndExpression* → *AndExpression* **&&** *Term* | *Term*

Se prevodi u:

(3) *AndExpression* → *Term AndExpression'*

(3.1) *AndExpression'* → **&&** Term *AndExpression'* | **ε**

Smene nakon uklanjanja levih rekurzija

(1) *RedoLoop* → **loop (** *Expression* **) {** *Statement* **redo (** *Expression* **) ;** *Statement* **}**

(2) *Expression* → *AndExpression Expression'*

(3) *Expression'* → **||** *AndExpression Expression'* | **ε**

(4) *AndExpression* → *Term AndExpression'*

(5) *AndExpression'* → **&&** Term *AndExpression'* | **ε**

(6) *Term* → **ID** | **CONST**

(7) *Statement* → *RedoLoop* | **ID =** *Expression* **;**

First

R. br.	Smena	FIRST (Smena)
1	*RedoLoop* → **loop** (*Expression*) { *Statement* **redo** (*Expression*) **;** *Statement* }	FIRST (**loop** (*Expression*) { *Statement* **redo** (*Expression*) **;** *Statement* }) = { **loop** }
2	*Expression* → *AndExpression Expression'*	FIRST (*AndExpression Expression'*) = FIRST (*Term AndExpression'*) U FIRST (**&&** Term *AndExpression'* | **ε**) = { **&&** , **ε** , **ID** , **CONST** }
3	*Expression'* → **||** *AndExpression Expression'* | **ε**	FIRST (**||** *AndExpression Expression'* | **ε**) = { **||** , **ε** }
4	*AndExpression* → *Term AndExpression'*	FIRST (*Term AndExpression'*) = { **ID** , **CONST** }
5	*AndExpression'* → **&&** Term *AndExpression'* | **ε**	FIRST (**&&** Term *AndExpression'* | **ε**) = { **&&** , **ε**)
6	*Term* → **ID** | **CONST**	FIRST (**ID** | **CONST**) = { **ID** , **CONST** }
7	*Statement* → *RedoLoop* | **ID =** *Expression* **;**	FIRST (*RedoLoop* | **ID =** *Expression* **;**) = { **loop** }

Follow

∪ ⊂ ∈

1.1 RedoLoop: FOLLOW(*RedoLoop*) = { **#** ,

∈ FOLLOW(*RedoLoop*)

(7) FOLLOW(*Statement*) ⊂ FOLLOW(*RedoLoop*)

2.1. Expression: FOLLOW(*Expression*) = { **)** , **;** }

(1) **)** ∈ FOLLOW(*RedoLoop*)

(7) **;** ∈ FOLLOW(*RedoLoop*)

3.1. Expression': FOLLOW(*Expression'*) = { **)** , **;** }

(2) FOLLOW(*Expression*) ⊂ FOLLOW(*Expression'*)

(3) FOLLOW(*Expression'*) ⊂ FOLLOW(*Expression'*)

4.1. AndExpression: FOLLOW(*AndExpression*) = { **||** }

(2) FIRST(*Expression'*) − { **ε** } ⊂ FOLLOW(*AndExpression*)

5.1. AndExpression': FOLLOW(*AndExpression'*) = { **||** }

(4) FOLLOW(*AndExpression*) ⊂ FOLLOW(*AndExpression'*)

(5) FOLLOW(*AndExpression'*) ⊂ FOLLOW(*AndExpression'*)

6.1. Term: FOLLOW(*Term*) = { **&&** }

(4) FIRST(*AndExpression'*) − { **ε** } ⊂ FOLLOW(*Term*)

7.1. Statement: FOLLOW(*Statement*) = { **}** }

(1) **}** ∈ FOLLOW(*Statement*)

Sintaksna tabela

Simbol na vrhu magacina	Ulazni simbol												
	loop	**(**	**)**	**{**	**redo**	**;**	**}**	**||**	**&&**	**ID**	**CONST**	**=**	**;**
RedoLoop	1												
Expression			{**ε**, 2.1.}			{**ε**, 2.1.}			2	2	2		
Expression'			{**ε**, 3.1.}			{**ε**, 3.1.}		3					
AndExpression										4	4		
AndExpression'								{**ε**, 5.1.}	5				
Term										6	6		
Statement	7												
loop	pop												
(		pop											
)			pop										
{				pop									
redo					pop								
;						pop							
}							pop						
||								pop					
&&									pop				
ID										pop			
CONST											pop		
=												pop	
;													acc