

# Contents

<b>1 Esercizio 3.2</b>	<b>1</b>
1.1 Calcolo di NULL, FIRST e FOLLOW . . . . .	1
1.2 Calcolo degli insiemi GUIDA . . . . .	2
1.3 Trasformazione in una grammatica LL(1) equivalente . . . . .	3
1.3.1 Produzione per for . . . . .	3
1.3.2 Produzione per if . . . . .	3
1.4 Grammatica LL(1) completa . . . . .	4

## 1 Esercizio 3.2

### 1.1 Calcolo di NULL, FIRST e FOLLOW

	NULL	FIRST	FOLLOW
$\langle \text{prog} \rangle$		{ assign, print, read, for, if, { }	{ \$ }
$\langle \text{statlist} \rangle$		{ assign, print, read, for, if, { }	{ EOF, } }
$\langle \text{statlistp} \rangle$	x	{ ; }	{ EOF, } }
$\langle \text{stat} \rangle$		{ assign, print, read, for, if, { }	{ ;, else, end, EOF, } }
$\langle \text{assignlist} \rangle$		{ [ }	{ ;, else, end, EOF, } }
$\langle \text{assignlistp} \rangle$	x	{ [ }	{ ;, else, end, EOF, } }
$\langle \text{idlist} \rangle$		{ ID }	{ ), [ }
$\langle \text{idlistp} \rangle$	x	{ , }	{ ), [ }
$\langle \text{bexpr} \rangle$		{ <, >, <=, >=, ==, <> }	{ ) }
$\langle \text{expr} \rangle$		{ +, -, *, /, NUM, ID }	{ ,, ;, to, +, -, *, /, ), NUM, ID }
$\langle \text{exprlist} \rangle$		{ +, -, *, /, NUM, ID }	{ ) }
$\langle \text{exprlistp} \rangle$	x	{ , }	{ ) }

## 1.2 Calcolo degli insiemi GUIDA

---

GUIDA( $\langle prog \rangle \rightarrow \langle statlist \rangle$ EOF)	{ assign, print, read, for, if, { }
GUIDA( $\langle statlist \rangle \rightarrow \langle stat \rangle \langle statlist \rangle$ )	{ assign, print, read, for, if, { }
GUIDA( $\langle statlist \rangle \rightarrow ; \langle stat \rangle \langle statlist \rangle$ )	{ ; }
GUIDA( $\langle statlist \rangle \rightarrow \varepsilon$ )	{ EOF, } }
GUIDA( $\langle stat \rangle \rightarrow \text{assign } \langle assignlist \rangle$ )	{ assign }
GUIDA( $\langle stat \rangle \rightarrow \text{print } ( \langle assignlist \rangle )$ )	{ print }
GUIDA( $\langle stat \rangle \rightarrow \text{read } ( \langle idlist \rangle )$ )	{ read }
GUIDA( $\langle stat \rangle \rightarrow \text{for } ( \text{ID} := \langle expr \rangle ; \langle bexpr \rangle ) \text{ do } \langle stat \rangle$ )	{ for }
GUIDA( $\langle stat \rangle \rightarrow \text{for } ( \langle bexpr \rangle ) \text{ do } \langle stat \rangle$ )	{ for }
GUIDA( $\langle stat \rangle \rightarrow \text{if } ( \langle bexpr \rangle ) \langle stat \rangle \text{ else } \langle stat \rangle \text{ end}$ )	{ if }
GUIDA( $\langle stat \rangle \rightarrow \text{if } ( \langle bexpr \rangle ) \langle stat \rangle \text{ end}$ )	{ if }
GUIDA( $\langle stat \rangle \rightarrow \{ \langle statlist \rangle \}$ )	{ { }
GUIDA( $\langle assignlist \rangle \rightarrow [ \langle expr \rangle \text{ to } \langle idlist \rangle ] \langle assignlist \rangle$ )	{ [ }
GUIDA( $\langle assignlist \rangle \rightarrow [ \langle expr \rangle \text{ to } \langle idlist \rangle ] \langle assignlist \rangle$ )	{ [ }
GUIDA( $\langle assignlist \rangle \rightarrow \varepsilon$ )	{ ;, else, end, EOF, } }
GUIDA( $\langle idlist \rangle \rightarrow \text{ID } \langle idlist \rangle$ )	{ ID }
GUIDA( $\langle idlist \rangle \rightarrow , \text{ID } \langle idlist \rangle$ )	{ , }
GUIDA( $\langle idlist \rangle \rightarrow \varepsilon$ )	{ ), ] }
GUIDA( $\langle bexpr \rangle \rightarrow < \langle expr \rangle \langle expr \rangle$ )	{ < }
GUIDA( $\langle bexpr \rangle \rightarrow > \langle expr \rangle \langle expr \rangle$ )	{ > }
GUIDA( $\langle bexpr \rangle \rightarrow \leq \langle expr \rangle \langle expr \rangle$ )	{ <= }
GUIDA( $\langle bexpr \rangle \rightarrow \leq \langle expr \rangle \langle expr \rangle$ )	{ <= }
GUIDA( $\langle bexpr \rangle \rightarrow == \langle expr \rangle \langle expr \rangle$ )	{ == }
GUIDA( $\langle bexpr \rangle \rightarrow <> \langle expr \rangle \langle expr \rangle$ )	{ <> }
GUIDA( $\langle expr \rangle \rightarrow + ( \langle exprlist \rangle )$ )	{ + }
GUIDA( $\langle expr \rangle \rightarrow - \langle expr \rangle \langle expr \rangle$ )	{ - }
GUIDA( $\langle expr \rangle \rightarrow * ( \langle exprlist \rangle )$ )	{ * }
GUIDA( $\langle expr \rangle \rightarrow / \langle expr \rangle \langle expr \rangle$ )	{ / }
GUIDA( $\langle expr \rangle \rightarrow \text{NUM}$ )	{ NUM }
GUIDA( $\langle expr \rangle \rightarrow \text{ID}$ )	{ ID }
GUIDA( $\langle exprlist \rangle \rightarrow \langle expr \rangle \langle exprlist \rangle$ )	{ +, -, *, /, NUM, ID }
GUIDA( $\langle exprlist \rangle \rightarrow , \langle expr \rangle \rightarrow \langle exprlist \rangle$ )	{ , }
GUIDA( $\langle exprlist \rangle \rightarrow \varepsilon$ )	{ ) }

---

### 1.3 Trasformazione in una grammatica LL(1) equivalente

La grammatica data non è LL(1) per via dei seguenti insiemi guida per la variabile  $\langle stat \rangle$ .

#### 1.3.1 Produzione per for

Dati gli insiemi guida non LL(1) della grammatica per **for**:

---

GUIDA( $\langle stat \rangle \rightarrow \text{for ( ID := } \langle expr \rangle ; \langle bexpr \rangle ) \text{ do } \langle stat \rangle$ )	{ for }
GUIDA( $\langle stat \rangle \rightarrow \text{for ( } \langle bexpr \rangle ) \text{ do } \langle stat \rangle$ )	{ for }

---

Fattorizzando la parte non comune ad ambo le produzioni introducendo una variabile  $\langle statc \rangle$ , otteniamo un nuovo insieme guida per  $\langle stat \rangle$  che risulta essere LL(1):

---

GUIDA( $\langle stat \rangle \rightarrow \text{for ( } \langle statc \rangle \langle bexpr \rangle ) \text{ do } \langle stat \rangle$ )	{ for }
GUIDA( $\langle statc \rangle \rightarrow \text{ID := } \langle expr \rangle ;$ )	{ ID }
GUIDA( $\langle statc \rangle \rightarrow \varepsilon$ )	{ <, >, <=, >=, ==, <> }

---

#### 1.3.2 Produzione per if

Dati gli insiemi guida non LL(1) della grammatica per **if**:

---

GUIDA( $\langle stat \rangle \rightarrow \text{if ( } \langle bexpr \rangle ) \langle stat \rangle \text{ else } \langle stat \rangle \text{ end}$ )	{ if }
GUIDA( $\langle stat \rangle \rightarrow \text{if ( } \langle bexpr \rangle ) \langle stat \rangle \text{ end}$ )	{ if }

---

Fattorizzando la parte non comune ad ambo le produzioni introducendo una variabile  $\langle statc \rangle$ , otteniamo un nuovo insieme guida per  $\langle stat \rangle$  che risulta essere LL(1):

---

GUIDA( $\langle stat \rangle \rightarrow \text{if ( } \langle bexpr \rangle ) \langle stat \rangle \langle statp \rangle \text{ end}$ )	{ if }
GUIDA( $\langle statp \rangle \rightarrow \text{else } \langle stat \rangle$ )	{ else }
GUIDA( $\langle statp \rangle \rightarrow \varepsilon$ )	{ end }

---

## 1.4 Grammatica LL(1) completa

---

GUIDA( $\langle prog \rangle \rightarrow \langle statlist \rangle \text{ EOF}$ )	{ assign, print, read, for, if, { }
GUIDA( $\langle statlist \rangle \rightarrow \langle stat \rangle \langle statlist \rangle$ )	{ assign, print, read, for, if, { }
GUIDA( $\langle statlist \rangle \rightarrow ; \langle stat \rangle \langle statlist \rangle$ )	{ ; }
GUIDA( $\langle statlist \rangle \rightarrow \varepsilon$ )	{ EOF, } }
GUIDA( $\langle stat \rangle \rightarrow \text{assign } \langle assignlist \rangle$ )	{ assign }
GUIDA( $\langle stat \rangle \rightarrow \text{print } ( \langle assignlist \rangle )$ )	{ print }
GUIDA( $\langle stat \rangle \rightarrow \text{read } ( \langle idlist \rangle )$ )	{ read }
GUIDA( $\langle stat \rangle \rightarrow \text{for } ( \langle state \rangle \langle bepr \rangle ) \text{ do } \langle stat \rangle$ )	{ for }
GUIDA( $\langle stat \rangle \rightarrow \text{if } ( \langle bepr \rangle ) \langle stat \rangle \langle statp \rangle \text{ end}$ )	{ if }
GUIDA( $\langle stat \rangle \rightarrow \{ \langle statlist \rangle \}$ )	{ { }
GUIDA( $\langle state \rangle \rightarrow \text{ID} := \langle expr \rangle ;$ )	{ ID }
GUIDA( $\langle state \rangle \rightarrow \varepsilon$ )	{ <, >, <=, >=, ==, <> }
GUIDA( $\langle statp \rangle \rightarrow \text{else } \langle stat \rangle$ )	{ else }
GUIDA( $\langle statp \rangle \rightarrow \varepsilon$ )	{ end }
GUIDA( $\langle assignlist \rangle \rightarrow [ \langle expr \rangle \text{ to } \langle idlist \rangle ] \langle assignlist \rangle$ )	{ [ }
GUIDA( $\langle assignlist \rangle \rightarrow [ \langle expr \rangle \text{ to } \langle idlist \rangle ] \langle assignlist \rangle$ )	{ [ }
GUIDA( $\langle assignlist \rangle \rightarrow \varepsilon$ )	{ ;, else, end, EOF, } }
GUIDA( $\langle idlist \rangle \rightarrow \text{ID } \langle idlist \rangle$ )	{ ID }
GUIDA( $\langle idlist \rangle \rightarrow , \text{ID } \langle idlist \rangle$ )	{ , }
GUIDA( $\langle idlist \rangle \rightarrow \varepsilon$ )	{ ), ] }
GUIDA( $\langle bepr \rangle \rightarrow < \langle expr \rangle \langle expr \rangle$ )	{ < }
GUIDA( $\langle bepr \rangle \rightarrow > \langle expr \rangle \langle expr \rangle$ )	{ > }
GUIDA( $\langle bepr \rangle \rightarrow <= \langle expr \rangle \langle expr \rangle$ )	{ <= }
GUIDA( $\langle bepr \rangle \rightarrow <= \langle expr \rangle \langle expr \rangle$ )	{ <= }
GUIDA( $\langle bepr \rangle \rightarrow == \langle expr \rangle \langle expr \rangle$ )	{ == }
GUIDA( $\langle bepr \rangle \rightarrow <> \langle expr \rangle \langle expr \rangle$ )	{ <> }
GUIDA( $\langle expr \rangle \rightarrow + ( \langle exprlist \rangle )$ )	{ + }
GUIDA( $\langle expr \rangle \rightarrow - \langle expr \rangle \langle expr \rangle$ )	{ - }
GUIDA( $\langle expr \rangle \rightarrow * ( \langle exprlist \rangle )$ )	{ * }
GUIDA( $\langle expr \rangle \rightarrow / \langle expr \rangle \langle expr \rangle$ )	{ / }
GUIDA( $\langle expr \rangle \rightarrow \text{NUM}$ )	{ NUM }
GUIDA( $\langle expr \rangle \rightarrow \text{ID}$ )	{ ID }
GUIDA( $\langle exprlist \rangle \rightarrow \langle expr \rangle \langle exprlist \rangle$ )	{ +, -, *, /, NUM, ID }
GUIDA( $\langle exprlist \rangle \rightarrow , \langle expr \rangle \rightarrow \langle exprlist \rangle$ )	{ , }
GUIDA( $\langle exprlist \rangle \rightarrow \varepsilon$ )	{ ) }

---