

Esercizio A

Si consider l'insieme di simboli terminali $T = \{a, b\}$.

Per ognuno dei linguaggi seguenti si dia una grammatica che lo generi

1. l'insieme delle stringhe della forma $a^n b^n$ con $n \geq 0$ (n volte a seguite da n volte b, e.g., "", "ab", "aabb", ...)
2. l'insieme delle stringhe con lo stesso numero di a e di b (e.g., "abba", "baaabb", "ab", "", ...)
3. l'insieme delle stringhe palindrome con a e b (e.g., "", "a", "b", "abba", "bbaabb", ...)

Esercizio B

Si consideri la grammatica CF definita a lezione

$\text{Exp} ::= \text{Num} \mid \text{Exp} '+' \text{Exp} \mid \text{Exp} '*' \text{Exp} \mid '(' \text{Exp} ')'$

$\text{Num} ::= '0' \mid '1'$

1. Usando la nozione di derivazione a uno o più passi, mostrare che
 1. La stringa "0*1+0" appartiene al linguaggio generato da Exp;
 2. La stringa "1+(" non appartiene al linguaggio generato da Exp.
2. Mostrare che esistono diversi modi per derivare da Exp in uno o più passi la stringa "0*1+0";
3. Mostrare che esistono diversi modi per derivare da Exp in uno o più passi la stringa "0*1+0" anche se a ogni passo si rimpiazza sempre il simbolo non-terminale che compare più a sinistra (left-most strategy).

Considera la seguente variante della grammatica vista a lezione

$\text{Exp1} ::= \text{Exp1} '+' \text{Exp1} \mid \text{Exp2}$

$\text{Exp2} ::= \text{Exp2} '*' \text{Exp2} \mid \text{Exp3}$

$\text{Exp3} ::= \text{Num} \mid '(' \text{Exp1} ')'$

$\text{Num} ::= '0' \mid '1'$

4. La stringa "0+1*0" quante derivazioni left-most ammette a partire da Exp1?
È derivabile a partire da Exp2?
5. La string "0*1*0" quante derivazioni left-most ammette a partire da Exp1?
È derivabile a partire da Exp2?
6. Quali differenze osservi rispetto alla formulazione precedente della grammatica?

Considera la seguente ulteriore variante

$\text{Exp1} ::= \text{Exp2} '+' \text{Exp1} \mid \text{Exp2}$

$\text{Exp2} ::= \text{Exp3} '*' \text{Exp2} \mid \text{Exp3}$

$\text{Exp3} ::= \text{Num} \mid '(' \text{Exp1} ')'$

$\text{Num} ::= '0' \mid '1'$

7. La stringa "0+1*0" quante derivazioni left-most ammette a partire da Exp1?
È derivabile a partire da Exp2?
8. La string "0*1*0" quante derivazioni left-most ammette a partire da Exp1?
È derivabile a partire da Exp2?
9. Quali differenze osservi rispetto all3 formulazioni precedenti della grammatica?

Esercizio C

1. Data la grammatica

$\text{Exp} ::= \text{Num} \mid \text{Exp} \text{'+'} \text{Exp} \mid \text{Exp} \text{'*'} \text{Exp} \mid \text{'(' Exp ')}$

$\text{Num} ::= \text{'0'} \mid \text{'1'}$

mostrare che esistono due diversi alberi di derivazione per la stringa "1*1*1" a partire da Exp.

2. Data la grammatica

$\text{Exp} ::= \text{Term} \mid \text{Exp} \text{'+'} \text{Term} \mid \text{'-'} \text{Exp}$

$\text{Term} ::= \text{'0'} \mid \text{'1'} \mid \text{'-'} \text{Term} \mid \text{'(' Exp ')}$

mostrare che è ambigua per Exp.

Esercizio D

Considera le seguenti grammatiche e per ognuna di queste

- se è ambigua, mostra una stringa con due alberi di derivazione
- se è non ambigua, spiega perché

1.

$\text{Exp1} ::= \text{Exp1} \text{'+'} \text{Exp1} \mid \text{Exp2}$

$\text{Exp2} ::= \text{Exp2} \text{'*'} \text{Exp2} \mid \text{Exp3}$

$\text{Exp3} ::= \text{Num} \mid \text{'(' Exp1 ')}$

$\text{Num} ::= \text{'0'} \mid \text{'1'}$

2.

$\text{Exp1} ::= \text{Exp1} \text{'+'} \text{Exp2} \mid \text{Exp2}$

$\text{Exp2} ::= \text{Exp2} \text{'*'} \text{Exp2} \mid \text{Exp3}$

$\text{Exp3} ::= \text{Num} \mid \text{'(' Exp1 ')}$

$\text{Num} ::= \text{'0'} \mid \text{'1'}$

3.

$\text{Exp1} ::= \text{Exp2} \text{'+'} \text{Exp1} \mid \text{Exp2}$

$\text{Exp2} ::= \text{Exp3} \text{'*'} \text{Exp2} \mid \text{Exp3}$

$\text{Exp3} ::= \text{Num} \mid \text{'(' Exp1 ')}$

$\text{Num} ::= \text{'0'} \mid \text{'1'}$

4.

$\text{Stmt} ::= \text{ID '=' Exp ';' } \mid \text{'if' '(' Exp ')'} \text{ Stmt } \mid \text{Stmt Stmt } \mid \text{'{' Stmt '}'}$

$\text{Exp} ::= \text{ID} \mid \text{NUM} \mid \text{BOOL}$

(dove ID, NUM, BOOL descrivono i linguaggi, rispettivamente, degli identificatori, dei literal numerici e dei literal booleani)

5.

$\text{Stmt1} ::= \text{Stmt2 Stmt1} \mid \text{Stmt2}$

$\text{Stmt2} ::= \text{'if' '(' Exp ')'} \text{ Stmt3 } \mid \text{Stmt3}$

$\text{Stmt3} ::= \text{ID '=' Exp } \mid \text{'{' Stmt1 '}'}$

$\text{Exp} ::= \text{ID} \mid \text{NUM} \mid \text{BOOL}$

6.

$\text{Exp1} ::= \text{Exp2} \mid \text{Exp1 '||' Exp2}$

$\text{Exp2} ::= \text{Exp3} \mid \text{Exp2 '&&' Exp3} \mid \text{'!' Exp2}$

$\text{Exp3} ::= \text{Bool} \mid \text{'(' Exp1 ')}$

$\text{Bool} ::= \text{'true'} \mid \text{'false'}$

Esercizio E

1. Data la grammatica ambigua

$\text{Exp} ::= \text{Num} \mid \text{Exp} \text{'+'} \text{Exp} \mid \text{Exp} \text{'*'} \text{Exp} \mid \text{'('} \text{Exp} \text{'})'}$

$\text{Num} ::= \text{'0'} \mid \text{'1'}$

trasformarla in equivalenti grammatiche non ambigue che definiscono le seguenti regole sintattiche:

- * ha precedenza su + e associa da sinistra, + associa da destra
- * ha precedenza su + e associa da destra, + associa da sinistra
- + ha precedenza su * e associa da sinistra, * associa da sinistra
- + ha precedenza su * e associa da sinistra, * associa da destra
- + ha precedenza su * e associa da destra, * associa da sinistra
- + ha precedenza su * e associa da destra, * associa da destra

2. Trasformare la seguente grammatica ambigua in una equivalente non ambigua dove lo statement condizionale ha la precedenza sulla composizione di statement Stmt Stmt.

$\text{Stmt} ::= \text{ID} \text{'='} \text{Exp} \text{';' } \mid \text{'if' '('} \text{Exp} \text{'')} \text{ Stmt} \mid \text{Stmt Stmt} \mid \text{'{'} \text{ Stmt} \text{'}'}$

$\text{Exp} ::= \text{ID} \mid \text{BOOL}$

3. Trasforma la seguente grammatica in una equivalente non ambigua dove le priorità sono date da

$\text{'!' } > \text{'\&\&' } > \text{'||' } > \text{'->'}$

'\&\&' e '||' associano a sinistra e '->' associa a destra

$\text{Exp} ::= \text{Bool} \mid \text{Exp} \text{'->' } \text{Exp} \mid \text{Exp} \text{'||' } \text{Exp}$

$\mid \text{Exp} \text{'\&\&' } \text{Exp} \mid \text{'!' } \text{Exp} \mid \text{'('} \text{Exp} \text{'}'}$

$\text{Bool} ::= \text{'true'} \mid \text{'false'}$