Corso di Linguaggi di Programmazione: Modulo 1
Esercitazione in classe di fine modulo (n.1)

1) Sia Bexp l'insieme delle espressioni booleane generate dalla seguente sintassi astratta in BNF

B ::= true | false | a | ~B | B and B

Definire le regole di semantica operazionale strutturata secondo la strategia di valutazione "Esterna Sinistra". (N.B: a è una variabile booleana, ovvero che può assumere valore true o false in uno store σ; di store ce ne possono essere solo 2, ovvero {(a, true)} e {(a, false)}).

2) Mostrare che la grammatica al punto 1) è ambigua.

3) Disambiguare la grammatica al punto 1), assumendo che l'operatore di negazione ~ leghi di più di (ovvero abbia precedenza su) and, e che and associ a sinistra.

4) Dimostrare che la grammatica al punto 3) genera un linguaggio non regolare. (Suggerimento: osserva che tutte le stringhe del tipo $(^n$ true $)^n$ appartengono a questo linguaggio).

5) Rimuovere la ricorsione sinistra nella grammatica prodotta al punto 3).

6) Verificare che la grammatica prodotta la punto 5) è LL(1).

7) Costruire la tabella di parsing LL(1) per tale grammatica.

8) Considerare la grammatica al punto 5) e produrne una equivalente in cui sono state rimosse le produzioni epsilon e quelle unitarie.

9) Considerare la grammatica al punto 5). Costruire per essa l'automa LR(0).

10) Verificare che la grammatica al punto 5) è SLR(1), ma non LR(0).

Risoluzione Esercitazione Fine Corso

1) $Var = \{a\}$ $\quad Store = \{\sigma \mid \sigma : Var \rightarrow \{true, false\}\}$

$\sigma_1 = \{(a, true)\}$ $\quad \sigma_2 = \{(a, false)\}$

$\Gamma = \{ \langle B, \sigma \rangle \mid B \in Bexp, \sigma \in Store \}$

$T = \{ \langle true, \sigma_1 \rangle, \langle true, \sigma_2 \rangle, \langle false, \sigma_1 \rangle, \langle false, \sigma_2 \rangle \}$

$\rightarrow \subseteq \Gamma \times \Gamma$ è la più piccola relazione generata dalle seguenti regole

$$\frac{}{\langle a, \sigma \rangle \rightarrow \langle \sigma(a), \sigma \rangle}$$

$$\frac{\langle B, \sigma \rangle \rightarrow \langle B', \sigma \rangle}{\langle \sim B, \sigma \rangle \rightarrow \langle \sim B', \sigma \rangle} \qquad \frac{}{\langle \sim true, \sigma \rangle \rightarrow \langle false, \sigma \rangle}$$

$$\frac{}{\langle \sim false, \sigma \rangle \rightarrow \langle true, \sigma \rangle}$$

$$\frac{\langle B_1, \sigma \rangle \rightarrow \langle B_1', \sigma \rangle}{\langle B_1 \text{ and } B_2, \sigma \rangle \rightarrow \langle B_1' \text{ and } B_2, \sigma \rangle}$$

$$\frac{}{\langle true \text{ and } B_2, \sigma \rangle \rightarrow \langle B_2, \sigma \rangle}$$

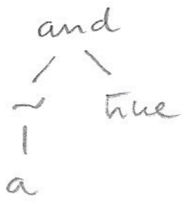
$$\frac{}{\langle false \text{ and } B_2, \sigma \rangle \rightarrow \langle false, \sigma \rangle}$$

2) La grammatica è ambigua. $B ::= true \mid false \mid a \mid \sim B \mid B \text{ and } B$

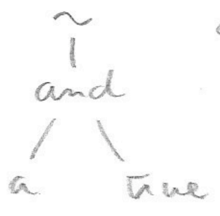
Ad esempio:

$w = \sim a \text{ and } true$

$B \Rightarrow B \text{ and } B \Rightarrow \sim B \text{ and } B \Rightarrow \sim a \text{ and } B \Rightarrow \sim a \text{ and } true$



$B \Rightarrow \sim B \Rightarrow \sim B \text{ and } B \Rightarrow \sim a \text{ and } B \Rightarrow \sim a \text{ and } true$



Due distinte derivazioni canoniche sx
⇓
2 diversi alberi di der.
⇓
2 diversi alberi sintattici

3/ - "~" lega di più di "and"

- "and" associa a sx

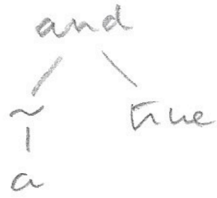
associa a sx

$B \rightarrow B \text{ and } T \mid T$

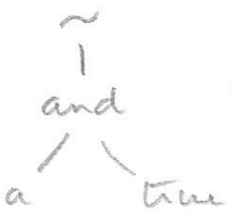
$T \rightarrow \sim T \mid A$ ← "~" lega più di "and"

$A \rightarrow \text{true} \mid \text{false} \mid a \mid (B)$

$B \Rightarrow B \text{ and } T \Rightarrow T \text{ and } T \Rightarrow \sim T \text{ and } T \Rightarrow \sim A \text{ and } T$
$\Rightarrow \sim a \text{ and } T \Rightarrow \sim a \text{ and } A \Rightarrow \sim a \text{ and true}$



$B \Rightarrow T \Rightarrow \sim T \Rightarrow \sim A \Rightarrow \sim (B) \Rightarrow \sim (B \text{ and } T)$
$\Rightarrow \sim (T \text{ and } T) \Rightarrow \sim (A \text{ and } T) \Rightarrow \sim (a \text{ and } T)$
$\Rightarrow \sim (a \text{ and } A) \Rightarrow \sim (a \text{ and true})$



albero di sintassi astratta

---

4) Il ling. non è regolare.
- Prendiamo N generico
- Scegliamo $z = (^N a)^N \in L$
- Per ogni uvw tali che $- z = uvw$ $- |uv| \le N$ $- |v| \ge 1$
  deve essere $v = (^i \quad 1 \le i \le N$
  $\Rightarrow uv^2w = (^{N+i} a)^N \notin L$

$\Rightarrow L$ non regolare

)

$B \to T B'$ $\qquad$ $T \to \sim T \mid A$

$B' \to \text{and } T B' \mid \varepsilon$ $\qquad$ $A \to \text{true} \mid \text{false} \mid a \mid (B)$

6) La grammatica ↑ è LL(1)?

	First	Follow
B	~, true, false, a, (	\$,)
B'	and, ε	\$,)
T	~, true, false a, (	and,), \$
A	true, false a, (	and,), \$

Casi da controllare:

(a) $B' \to \text{and } T B' \mid \varepsilon$

(1) $\text{First}(\text{and } T B') = \{\text{and}\}$
$\cap\ \text{First}(\varepsilon) = \{\varepsilon\} = \emptyset$

(2) $\text{First}(\text{and } T B') = \{\text{and}\}$
$\cap\ \text{Follow}(B') = \{\$,)\} = \emptyset$

$\Rightarrow$ ok

(b) $T \to \sim T \mid A$

(1) $\text{First}(\sim T) = \{\sim\} \cap$
$\text{First}(A) = \{\text{true}, \text{false}, a, (\} = \emptyset$

$\Rightarrow$ ok

(c) $A \to \text{true} \mid \text{false} \mid a \mid (\varepsilon)$

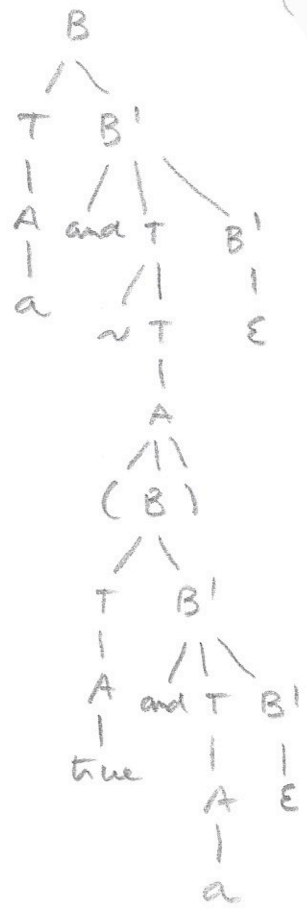
tutti i First sono diversi
$\Rightarrow$ ok

$\Rightarrow$ la gr. è LL(1)

7)

	a	true	false	(	)	and	~	\$
B	$B \to TB'$	$B \to TB'$	$B \to TB'$	$B \to TB'$			$B \to TB'$	
B'					$B' \to \varepsilon$	$B' \to \text{and}TB'$		$B' \to \varepsilon$
T	$T \to A$	$T \to A$	$T \to A$	$T \to A$			$T \to \sim T$	
A	$A \to a$	$A \to \text{true}$	$A \to \text{false}$	$A \to (B)$				

B	a and ~ (true and a) $
T B'	a ...
A B'	a ...
$\cancel{a}$ B'	$\cancel{a}$ and ~ (true and a) $
B'	and ~ ...
$\cancel{\text{and}}$ T B'	$\cancel{\text{and}}$ ~ (true and a) $
T B'	~ (true and a) $
$\cancel{\sim}$ T B'	$\cancel{\sim}$ (
T B'	(true and a) $
A B'	(
$\cancel{(}$ B) B'	$\cancel{(}$ true ...
B) B'	true and a) $
T B') B'	true ...
A B') B'	true ...
$\cancel{\text{true}}$ B') B'	$\cancel{\text{true}}$ and a) $
B') B'	and a) $
$\cancel{\text{and}}$ T B') B'	$\cancel{\text{and}}$ a) $
T B') B'	a) $
A B') B'	a) $
$\cancel{a}$ B') B'	$\cancel{a}$) $
B') B'	) $
$\cancel{)}$ B'	$\cancel{)}$ $
B'	$
ε	$



8) Presa la grammatica LL(1)

$B \to T B'$ $\qquad$ $T \to \sim T \mid A$

$B' \to \text{and } T B' \mid \varepsilon$ $\qquad$ $A \to \text{true} \mid \text{false} \mid a \mid (B)$

togliere la prod. ε e le prod. unitarie (in questo ordine).

$N(G) = \{B'\}$ Tutte le prod. che contengono B' nella parte destra vanno sdoppiate

$B \to T B' \mid T$ ← nuova prod. unitaria

$B' \to \text{and } T B' \mid \text{and } T$

$T \to \sim T \mid A$

$A \to \text{true} \mid \text{false} \mid a \mid (B)$

Ora togliere le 2 prod. unitarie

$U(G) = \underbrace{Id}_{0\ \text{round}} \cup \{ \underbrace{(B,T), (T,A)}_{1°\ \text{round}}, \underbrace{(B,A)}_{2°\ \text{round}} \}$

$B \to T B' \mid \sim T \mid \text{true} \mid \text{false} \mid a \mid (B)$

$B' \to \text{and } T B' \mid \text{and } T$

$T \to \sim T \mid \text{true} \mid \text{false} \mid a \mid (B)$

$A \to \text{true} \mid \text{false} \mid a \mid (B)$

9) Automa LR(0) per

(0) $S \to B$

(1) $B \to T B'$

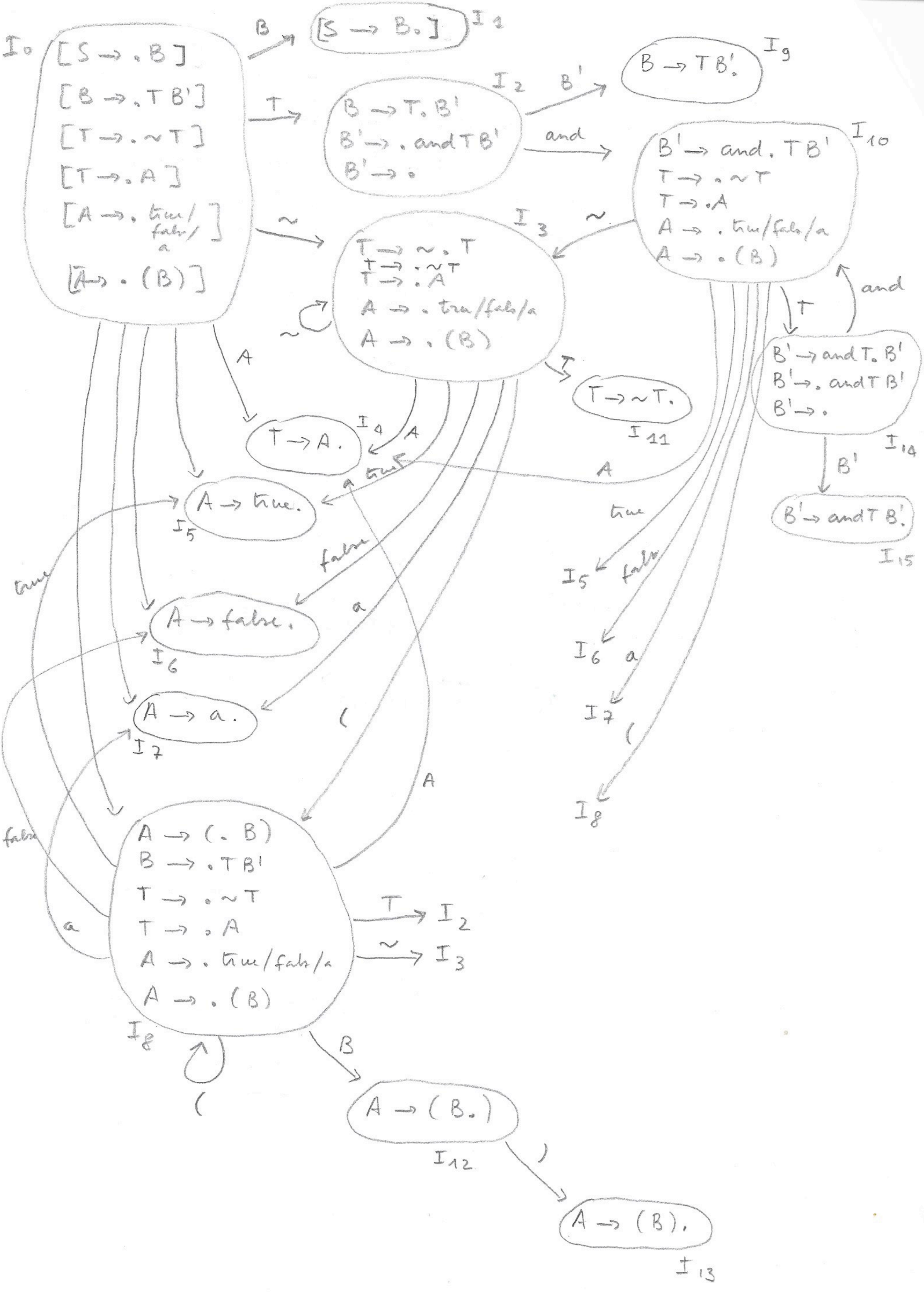
(2) $B' \to \text{and } T B'$

(3) $B' \to \varepsilon$

(4-5) $T \to \sim T \mid A$

(6-9) $A \to \text{true} \mid \text{false} \mid a \mid (B)$

	First	Follow
B	~, true, false, a, (	\$,)
B'	and, ε	\$,)
T	~, true, false, a, (	and, \$,)
A	true, false, a, (	and, \$,)



I0
[S → . B]
[B → . T B']
[T → . ~ T]
[T → . A]
[A → . true/false/a]
[A → . (B)]
I1
[S → B .]
I2
B → T . B'
B' → . and T B'
B' → .
I9
B → T B' .
I10
B' → and . T B'
T → . ~ T
T → . A
A → . true/false/a
A → . (B)
I3
T → ~ . T
T → . ~ T
T → . A
A → . true/false/a
A → . (B)
I11
T → ~ T .
I4
T → A .
I5
A → true .
I6
A → false .
I7
A → a .
I14
B' → and T . B'
B' → . and T B'
B' → .
I15
B' → and T B' .
I8
A → (. B)
B → . T B'
T → . ~ T
T → . A
A → . true/false/a
A → . (B)
I12
A → (B .)
I13
A → (B) .
B
T
B'
and
~
A
true
false
a
(
)
I2
I3
I5
I6
I7
I8

	a	true	false	and	~	(	)	$	B	B'	T	A
I_0	s7	s5	s6		s3	s8			g1		g2	g4
I_1								acc				
I_2				s10			r3	r3		g9		
I_3	s7	s5	s6		s3	s8					g11	g4
I_4				r5			r5	r5				
I_5				r6			r6	r6				
I_6				r7			r7	r7				
I_7				r8			r8	r8				
I_8	s7	s5	s6		s3	s8			g12		g2	g4
I_9							r1	r1				
I_{10}	s7	s5	s6		s3	s8					g14	g4
I_{11}				r4			r4	r4				
I_{12}							s13					
I_{13}				r9			r9	r9				
I_{14}				s10			r3	r3		g15		
I_{15}							r2	r2				

Follow (B') = { $,) }

(3) B' → ε

(2) B' → and T B'

Follow (A) = { and ,), $ }

(6-9) A → true / false / a / (B)

Follow (T) = { and ,), $ }

(5) T → A

(4) T → ~T

Follow (B) = {), $ }

(1) B → T B'

0 ε a and (~ true) $

~~7~~ a A and (~ true) $ A → a

0

~~4~~ A T and (~ true) $ T → A

0

2 T and ...

10 T and (~ true) $

8 T and (~ true) $

3 T and (~ true) $

~~5~~ T and (~ true A) $ A → true

0

2

10

8

3

~~4~~ T and (~ A T) $ T → A

0

2

10

8

~~3~~

~~11~~ T and (~ T T) $ T → ~T

0

2

10

8

~~2~~ T and (T B' ε) $ B' → ε

~~9~~ T and (T B' B) $ B → T B'

0

2

10

~~8~~

~~12~~ T and (B) $

~~13~~ T and (B) A $ A → (B)

0

2

10

~~4~~ T and A T $ T → A

0

2

~~10~~

~~14~~ T and T B' $ B' → ε

~~15~~ T and T B' B' $ B' → and T B'

0

~~2~~

~~9~~ T B' B $ B → T B'

0

1 B $ accept