

# Laurea Magistrale in INFORMATICA

## Principi di Linguaggi di Programmazione

### Compilatori

prof. M. Bellia  
Appello VI - 5 settembre 2013

(tempo a disposizione: 2 ore – Totalizzare almeno la metà dei punti in ciascun esercizio)

**Esercizio 1.** (punti 3 – 6) Si consideri la seguente espressione regolare:  $E \equiv (alb)^* a (alb)^* b$ .

- Si dia il dotted automata di E (mostrando il calcolo e gli items di ogni stato);
- Utilizzando l'algoritmo di minimizzazione si mostri come si prova se tale automa è minimo.

**Esercizio 2.** (punti 7 - 13) Sia  $L = \{a^{n_1} b^{m_1} c^{k_1} \dots a^{n_k} b^{m_k} c^{k_k} \mid n_1, m_1, \dots, n_k, m_k \geq 0, k \geq 0\}$  un linguaggio, e  $L_s$  il suo sottolinguaggio così definito  $L_s = \{a^{n_1} b^{m_1} c^{k_1} \dots a^{n_k} b^{m_k} c^{k_k} \mid n_1, \dots, n_k > 0, k > 0\}$

- Si dia una grammatica e una tabella di analisi LALR per L
- Si dia un analizzatore lineare per  $L_s$
- Si mostri il comportamento dell'analizzatore in b, durante l'analisi di: cc.

**Esercizio 3.** (punti 5 - 11) Si aggiunga al linguaggio Semplice, un costrutto iterativo della seguente forma:  
**iter E in C1... Cn ni**

dove E è un'espressione intera e  $C_i$ , per  $1 \leq i \leq n$ , sono comandi che formano una sequenza non vuota. La semantica prevede che E sia valutato ad ogni iterazione e il valore risultante k sia utilizzato per selezionare il comando  $C_k$  se  $1 \leq k \leq n$ . In tale caso, prima è eseguito  $C_k$ , dopo si procede con una nuova iterazione del costrutto. L'iterazione termina quindi, quando e se  $k \notin [1..n]$ .

- Si forniscano opportune produzioni per estendere la grammatica di Semplice, in particolare il non terminale C dei comandi. Le produzioni devono estendere in modo non ambiguo e adatto ad un oblivious bottom-up.
- Si mostri che la grammatica riconosce il seguente comando:  
iter x in iter x in y:=y\*y ni; x:=1; x:=1; x:=1; ni
- Si estendano tali produzioni in uno schema di traduzione, ascendente, oblivious per la generazione di codice 3AC di costrutti **iter\_in\_ni**.