

Linguaggi di Programmazione e Compilatori

III° Appello del 9/9/2004

Scrivere **in stampatello** COGNOME e NOME su ogni foglio. La brutta copia va consegnata (indicare che si tratta della brutta), il testo no.

NOTA: Nelle espressioni regolari si possono usare le usuali convenzioni di precedenza: l'operatore $*$ lega più della concatenazione che, a sua volta, lega più dell'operatore $|$. Inoltre si può usare l'abbreviazione $+$ con il solito significato.

ESERCIZIO 1 (7 punti)

Si scriva una definizione regolare che definisca una categoria lessicale **commands** che denoti il linguaggio così definito:

- Ogni comando inizia con il carattere $\backslash$ o con il carattere $!$
- Dopo il carattere di inizio deve essere inserito, senza spazi intermedi, il nome del comando che deve iniziare per lettera e può proseguire con lettere o cifre
- Il nome del comando può essere seguito da un parametro opzionale che deve essere inserito fra parentesi graffe dopo il nome del comando senza spazi intermedi. Il parametro può essere un qualunque numero intero che non comincia per zero (e quindi non può essere nemmeno zero).

Esempi di comandi validi: $\backslash\text{com}1$, $!c$, $\backslash\text{fun}\{67\}$. Suggerimento: si ricordi che nelle definizioni regolari possono essere definiti dei nomi per denotare espressioni regolari di comodo (es. **letter**).

ESERCIZIO 2 (12 punti)

Si consideri il seguente linguaggio:

$$\{a^n b c^{n+1} \mid n > 0\} \cup \{b^n c d^k \mid n \geq 0, k > 0\} \cup \{d^m a c^k b^m \mid m = n + k, n > 0, k > 0\}$$

1. Si scriva una grammatica libera dal contesto per il linguaggio
2. Il linguaggio è LL(1)? Se sì, si scriva la tabella per un parser predittivo top-down che analizzi il linguaggio.
3. Utilizzando la tabella si faccia il parsing delle stringhe *aabccc* e *cddd*

ESERCIZIO 3 (12 punti)

Si consideri la seguente grammatica

$$\begin{aligned} S &\rightarrow Ad \mid Bbc \\ A &\rightarrow aB \\ B &\rightarrow aAb \mid c \mid d \end{aligned}$$

1. Si scriva il linguaggio generato dalla grammatica utilizzando espressioni su insiemi
2. La grammatica è LR(1)? Se sì, si scriva la tabella per un parser bottom-up shift-reduce che analizzi il linguaggio.
3. Si dica quali sono tutti gli item validi per il viable prefix *aaaaa*