

Tempo a disposizione: ore 2.

Svolgere gli esercizi 1-4 e 5-8 su due fogli differenti.

Per Paradigmi: svolgere solo: 2,5,6,7,8.

1. Si consideri la seguente grammatica lineare-destra G con simbolo iniziale S :

$$\begin{aligned} S &\rightarrow \mathbf{a}B \mid \mathbf{a}C \mid \mathbf{a} \\ B &\rightarrow \mathbf{a}S \\ C &\rightarrow \mathbf{a}S \end{aligned}$$

(i) Costruisci l'NFA associato a G , seguendo la costruzione vista a lezione. (ii) Quindi costruisci il DFA associato a tale NFA, attraverso la costruzione per sottoinsiemi. (iii) Ricava dal DFA così ottenuto la grammatica lineare-destra associata. (iv) Determina l'espressione regolare associata a tale grammatica.

2. Si consideri il linguaggio $L = \{a^i b^j c^k \mid i \neq j \vee j \neq k, \text{ con } i, j, k \geq 0\}$. (i) La stringa vuota ϵ appartiene a L ? (ii) Elenca tutte le stringhe $w \in L$ tali che $|w| \leq 3$. (iii) Definisci una grammatica libera G tale che $L(G) = L$, aggiungendo alcune produzioni per S alle seguenti:

$$\begin{aligned} S &\rightarrow A \mid B \mid C \mid AB \mid BC \mid ATC \\ A &\rightarrow \mathbf{a} \mid \mathbf{a}A \\ B &\rightarrow \mathbf{b} \mid \mathbf{b}B \\ C &\rightarrow \mathbf{c} \mid \mathbf{c}C \\ T &\rightarrow \mathbf{a}T\mathbf{b} \mid \mathbf{ab} \\ V &\rightarrow \mathbf{b}V\mathbf{c} \mid \mathbf{bc} \end{aligned}$$

(iv) Tale grammatica G é di classe $LL(1)$?

3. Si consideri la grammatica G

$$S \rightarrow x \mid y \mid S * S \mid (S)$$

che esprime espressioni di moltiplicazione costruite con le variabili intere x e y . Definire le regole di semantica operativa strutturata (SOS) per valutare tali espressioni secondo la regola esterna sinistra, supponendo che $\sigma : \{x, y\} \rightarrow \mathbb{N}$ sia uno store per le variabili.

4. Si consideri la grammatica G del punto precedente. (i) Mostrare che G è ambigua. (ii) Trovare una grammatica G' equivalente ma non ambigua, supponendo che l'operatore $*$ associ a destra. (iii) Verificare se G' sia di classe $LL(1)$. Se non lo fosse, manipolarla opportunamente (ad esempio, fattorizzando) per ottenere una grammatica equivalente G'' di classe $LL(1)$. (iv) Costruire il parser $LL(1)$ e verificarne il comportamento sull'input $(x * y) * x$.

5. Si scriva una dichiarazione di funzione e un frammento di codice con una chiamata alla stessa tale che, passando i parametri per nome oppure per valore, si producano output diversi.
6. Un certo linguaggio adotta un'allocazione della memoria completamente statica. Quali delle seguenti caratteristiche *non sono compatibili* con tale politica di allocazione? (a) Blocchi annidati; (b) Definizione di funzioni annidate; (c) Definizione di funzioni ricorsive; (d) Passaggio dei parametri per valore; (e) Passaggio dei parametri per riferimento; (d) comandi allocazione esplicita della memoria.
7. In un certo linguaggio si incontra l'istruzione

```
write(h(1) + h(1.2))
```

e si osserva che stampa 2. Qual è il meccanismo più semplice di cui deve essere dotato tale linguaggio perché ciò sia possibile? Si diano possibili definizioni per `h`.

8. Si consideri la seguente definizione in uno pseudolinguaggio che ammette ricorsione e dove il passaggio dei parametri avviene per nome:

```
int f(int n, int m){
    if (n==0) return m+10;
    else return f(n-1, m++);
}
int x = 1;
y= f(3,x);
write(x)
```

Il codice è tradotto con il migliore compilatore ottimizzante presente sul mercato. Quanti RdA per `f` devono essere messi sulla pila di sistema per il calcolo di `f(3,x)`? Qual è il valore stampato ?