

Tempo a disposizione: ore 2.

**SCRIVERE LE SOLUZIONI A 1-4 E 5-8 SU DUE FOGLI DIVERSI**

1. Con la notazione  $\mathcal{C}_{L_1, L_2}^L$  indichiamo un compilatore da  $L_1$  a  $L_2$  scritto in  $L$ . Con  $\mathcal{I}_{L_1}^L$  indichiamo un interprete scritto in  $L$  per il linguaggio  $L_1$ . Infine se  $P^L$  è un programma scritto in  $L$  e  $x$  un suo dato di input,  $\mathcal{I}_{L_1}^L(P^L, x)$  indica l'applicazione dell'interprete a  $P^L$  e  $x$ . Si dica se la seguente scrittura ha senso

$$\mathcal{I}_{L_1}^L(\mathcal{I}_{L_1}^L, (\mathcal{C}_{L, L_1}^L, P^L)).$$

Se la risposta è “no” si motivi tale fatto; se è “sì”, si dica qual è il risultato ottenuto e se tale risultato sia ottenibile in modo più semplice (sotto le stesse ipotesi che permettono di ottenerlo con l'espressione precedente).

2. Applicando la costruzione canonica, si dia un NFA per il linguaggio dell'espressione regolare

$$a(b + bc^+)^*c$$

3. Si dia un PDA per il linguaggio

$$\{a^{2n}b^na^m \mid n, m \geq 0\}$$

specificando la modalità di accettazione.

4. Si consideri la grammatica (simbolo iniziale  $S$ ; solite convenzioni per terminali/non terminali):

$$(1) \quad S \rightarrow aDC \mid C$$

$$(2) \quad B \rightarrow aBb \mid b$$

$$(3) \quad C \rightarrow cS \mid \varepsilon$$

$$(4) \quad D \rightarrow bD \mid \varepsilon$$

Si dica se si tratta di una grammatica LL(1).

5. Si dica cosa viene stampato dal seguente frammento di codice scritto in uno pseudo-linguaggio che usa scoping dinamico e passaggio di parametri per nome.

```
int x=2
void foo (nome int y){
    x = x + 1;
    y = y + 20;
    x = x + y;
    write (x);
}
{ int x=50
  foo (x);
  write (x);
}
```

6. Si dica cosa viene stampato dal seguente frammento di codice assumendo che il passaggio dei parametri sia per riferimento e lo scope statico.

```
{int y=0;
void f(int x){
    x = x+1;
    if y = 1 throw E;
    x = x+1;
}
void g(int x){
    try{ f(x); } catch E { write (y+10)}
}
try{ g(y); } catch E { write (y+20)};
write(y+30);
}
```

7. In un linguaggio che permette overloading, si possono scrivere le seguenti quattro espressioni

```
1+2
1.0+2.0
1+2.0
1.0+2
```

Com'è possibile che tutte e quattro le espressioni siano corrette, sapendo che + ha due soli significati sovraccaricati?

8. Si consideri un linguaggio con scope statico, implementato mediante display, nel quale gli identificatori sono noti staticamente. Si assuma inoltre che ogni nome usato sia dichiarato in un solo blocco. Si prendano in considerazione le operazioni di (NL)=“accesso ad una variabile non locale x” e (L)=“accesso ad una variabile locale y” (nel contesto di un blocco). Per ognuna delle due operazioni di dica a quale dei seguenti parametri è proporzionale il tempo necessario all'esecuzione dell'operazione, motivando brevemente la risposta.

(i) Il numero di variabili presenti nel programma; (ii) Il numero di variabili presenti nei blocchi compresi tra quello di dichiarazione della variabile e quello in cui si accede alla variabile stessa; (iii) Il numero di record di attivazione presenti sulla pila e compresi tra quello contenente la dichiarazione della variabile e quello in cui vi si accede; (iv) il numero di blocchi che contengono testualmente il blocco in cui si accede alla variabile e che sono contenuti in quello nel quale la variabile è dichiarata; (v) Il tempo è costante, e quindi indipendente da questi parametri; (vi) nessuna delle risposte precedenti.