

Tempo a disposizione: ore 2.

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

**Per Paradigmi: svolgere solo: 1,5,6,7,8.**

1. Dimostrare che il linguaggio  $L_1 = \{a^{n^3} \mid n \geq 0\}$  non è regolare.
2. Costruire un DPDA che riconosca il linguaggio  $L_2 = \{a^n c b^n \mid n \geq 0\}$ .
3. Costruire un parser  $LL(1)$  per il linguaggio  $L_3 = \{a^n c b^m \mid 0 \leq n \leq m\}$  e si discuta il suo comportamento sugli input  $acbb$  e  $c$ . (Suggerimento: per generare una grammatica di classe  $LL(1)$ , osservate che  $L_3$  si ottiene concatenando il linguaggio  $L_2$ , la cui grammatica è ovvia, con il linguaggio denotato dall'espressione regolare  $b^*$ , la cui grammatica è altrettanto ovvia.)
4. Si consideri il linguaggio  $L_4 = \{w \in (0|1)^* \mid |w| \geq 2 \text{ ed il penultimo simbolo di } w \text{ è } 1\}$ . Si descriva un'espressione regolare che denoti  $L$ . Si produca un DFA che riconosca  $L_4$  (non è necessario partire dall'espressione regolare).

5. Si fornisca un frammento di codice (anche in pseudo-linguaggio) che generi un “dangling reference”. Si faccia quindi vedere come con la tecnica delle “tombstones” non si ha più tale problema.
6. Si consideri un linguaggio con scope statico, implementato mediante display, nel quale tutti gli identificatori sono noti staticamente. Si prendano in considerazione le operazioni di “accesso ad una variabile non locale x” e “accesso ad una variabile locale y” (nel contesto di un blocco). Per ognuna delle due operazioni si 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 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.
7. Si dica cosa stampa il seguente frammento in uno pseudolinguaaggio con passaggio per nome e scope statico.

```

int x = 10;
int y = 0;
void foo(name int x, name int z){
    int y = 5;
    int v = 100;
    v=x;
    if (x==y) write(y);
    else write (x);
    if (x==z) write(x);
    else write (y);
}
foo(y++,y);
write(x);
write(y);

```

8. Si considerino le seguenti definizioni Java

```

interface I{
    void f();
    int s(int y);
}
interface J{
    void g();
}
class A implements I{
    int x;
    void f(){
        x++;
    }
    int s(int y){
        return x+x;
    }
}
class B extends A implements J{
    void g(){
        x--;
    }
    void f(){
        x--;
    }
}
A a = new A();
B b = new B()
a.x = 10;
b.x = 20;
a = b;
System.out.println(a.x);
System.out.println(b.x);
a.f();
System.out.println(a.x);
System.out.println(b.x);

```

- (i) Cosa eredita B da A? Cosa eredita da I? E cosa da J?
- (ii) Cosa stampa il frammento?