

Tempo a disposizione: ore 2.

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

1. Si dimostri che le espressioni regolari $(a | \epsilon)^+ | a^*$ e $(\emptyset | a)^*$ denotano lo stesso linguaggio.
2. Si dimostri che il linguaggio $L = \{a^n b^m a^n b^m | n, m \geq 0\}$ non è libero.
3. Si consideri la seguente grammatica G con simbolo iniziale S :

$$\begin{aligned} S &\rightarrow aS | bS | c | A \\ A &\rightarrow Ac | c \end{aligned}$$

(i) Quale linguaggio genera G ? (ii) Mostrare che la grammatica G è ambigua e la si disambigui.
(iii) Dato un linguaggio regolare L , esiste sempre una grammatica non ambigua che lo genera?
Ovvero possono esistere linguaggi regolari inerentemente ambigui?

4. Si consideri la grammatica G con simbolo iniziale S :

$$\begin{aligned} S &\rightarrow bSCa | c \\ C &\rightarrow \epsilon | cC \end{aligned}$$

(i) Costruire l'automa canonico LR(0). (ii) Costruire la tabella di parsing SLR(1) e verificare se ci sono conflitti. (iii) Mostrare il funzionamento del parser SLR(1) per l'input *bcca*.

5. Si dica, motivando la risposta, cosa viene stampato dal seguente frammento di codice scritto in un pseudo-linguaggio che usa scoping statico e passaggio di parametri per valore e per nome.

```
int x = 1;
int y = 2;
int z = 3;
void pippo(value int x, name int y){
    x = x + x;
    y = y + y;
    z = z + z;
    write(x,y,z)
}
{ int x = 10;
  int y = 20;
  int z = 30;
  pippo(x++, y++);
  write(x,y,z);
}
write(x,y,z));
```

(la primitiva `write(x)` permette di stampare un valore intero; un comando della forma `foo(w++)`; passa a `foo` il valore corrente di `w` e poi incrementa `w` di uno).

6. Si assuma che in un generico linguaggio imperativo a blocchi, il blocco A contenga una chiamata della funzione f . Il numero dei record di attivazione (RdA) presenti a run-time sulla pila fra il RdA di A e quello della chiamata di f è fissato staticamente o può variare dinamicamente? Motivare la risposta.

7. L'esecuzione del seguente programma Java

```
1 class A{
2 }
3
4 class B {
5 }
6
7 public class HelloWorld{
8
9     public static void main(String []args){
10         A[] aa = new A[10];
11         Object[] oo = aa;
12         oo[0] = new B();
13
14         System.out.println("Hello World");
15     }
16 }
```

produce il seguente messaggio

```
Exception in thread main java.lang.ArrayStoreException: B
at HelloWorld.main(HelloWorld.java:12)
```

Spiegare cosa è successo e perché.

8. Si dica cosa viene stampato dal seguente frammento di codice scritto in uno pseudo-linguaggio che permette il passaggio di funzioni come parametro, usa shallow binding, scoping dinamico e passaggio per valore.

```
int f( int function g (value int n), value int k) {
    int somma(value int z){
        return z+k;
    };
    int x = 10;
    int y = 10;

    if k = 1 return g(x)+g(y) else
    {int x = 30;
     int y = 30;
     return f (somma, k-1)
    }
}
int foo(value int n){
}
int x = 50;
int y = 50;
x= f(foo,2);
write(x);
```