

Tempo a disposizione: ore 2.

Svolgere gli esercizi 1–4, 5–6 e 7–8 su tre fogli separati.

Scrivere nome, cognome e matricola su ogni foglio consegnato.

- FOGLIO 1 ▷ 1. Siano date due classi di linguaggi formali A e B , tali che $A \subseteq B$. Si chiede se le seguenti due affermazioni siano vere. Giustificare la risposta. (Suggerimento: pensate alle classi dei linguaggi regolari, o liberi deterministici, o liberi.)

(a) Se A è chiusa rispetto all'operazione di unione, allora anche B lo è.

(b) Se B è chiusa rispetto all'operazione di unione, allora anche A lo è.

- FOGLIO 1 ▷ 2. Si consideri il seguente NFA $M = (\Sigma, Q, \delta, q_0, F)$, dove $\Sigma = \{a\}$, $Q = \{q_0, q_1, q_2, q_3, q_4, q_5\}$, $F = \{q_2, q_3\}$ e la funzione di transizione $\delta : Q \times (\Sigma \cup \{\epsilon\}) \rightarrow \mathcal{P}(Q)$ è così definita: $\delta(q_0, a) = \{q_1, q_2\}$, $\delta(q_1, a) = \{q_4\}$, $\delta(q_2, a) = \emptyset$, $\delta(q_3, a) = \emptyset$, $\delta(q_4, a) = \{q_1, q_5\}$, $\delta(q_5, a) = \emptyset$, $\delta(q_1, \epsilon) = \{q_3\}$ e, infine, $\delta(q, \epsilon) = \emptyset$ per $q \in \{q_0, q_2, q_3, q_4, q_5\}$.

(i) Si fornisca una rappresentazione grafica di M . (ii) Si determini il DFA M' minimo associato a M . (iii) Qual è il linguaggio riconosciuto da M' ? (iv) Fornire una espressione regolare che rappresenti tale linguaggio.

- FOGLIO 1 ▷ 3. Si consideri la grammatica G con simbolo iniziale S :

$$\begin{aligned} S &\rightarrow AB \\ A &\rightarrow \mathbf{aAb} \mid \epsilon \\ B &\rightarrow \mathbf{bB} \mid \epsilon \end{aligned}$$

(i) Verificare che G è di classe LL(1). (ii) Riempire la tabella di parsing LL(1). (iii) Mostrare il funzionamento del parser LL(1) per gli input abb e ϵ .

- FOGLIO 1 ▷ 4. Si consideri la seguente grammatica G con simbolo iniziale S :

$$\begin{aligned} S &\rightarrow \mathbf{aS a} \mid \mathbf{aBa} \\ B &\rightarrow \mathbf{bB} \mid \mathbf{c} \end{aligned}$$

(i) Si verifichi se G sia di classe SLR(1). (ii) Che linguaggio genera G ?

- FOGLIO 2 ▷ 5. Un certo linguaggio adotta un'allocazione della memoria dinamica con pila ma senza heap. Quali delle seguenti caratteristiche *sono compatibili* con tale politica di allocazione? (a) Blocchi annidati; (b) definizione di funzioni annidate; (c) passaggio dei parametri per riferimento; (d) comandi di allocazione esplicita della memoria; (e) creazione di strutture dati di dimensione variabile dinamicamente; (f) definizione di funzioni ricorsive; (g) Definizione di funzioni ricorsive in coda.

- FOGLIO 2 ▷ 6. Si dica cosa viene stampato dal seguente frammento di codice scritto in uno pseudo-linguaggio che usa scoping statico e deep binding (la procedura $write_integer(x)$ stampa il contenuto della variabile intera x):

```

int x = 3;
procedure stampa
  {write_integer(x);
  }
procedure assegna(function S; n:int)
  {x = n;
  S;
  }
procedure pippo(function S; function P; int n )
  {int x= 10;
  if n=1 then assegna(P; n) else S(n);
  stampa;
  }

{
  int x = 30;
  procedure assegna(function S; n:int)
    {x = n;
    S;
    }
  pippo(assegna, stampa, 1);
  write_integer(x)
  pippo(assegna, stampa, 2);
  write_integer(x)
}

```

FOGLIO 3

- ▷ 7. Viene dato un linguaggio con tipaggio statico e tipi base come `string` e `int`, prodotto $A \times B$ e somma $A + B$. Il linguaggio usa passaggio per riferimento senza riferimenti nulli. Si descrivano i due tipi `Persona` e `Oggetto`. `Persona` ha un `nome`, può avere una lista di figli, di tipo `Persona`, e può avere una lista di averi di tipo `Oggetto`. `Oggetto` ha un `nome` e può avere una lista di proprietari, di tipo `Persona`. Oltre alla definizione dei tipi, scrivere il codice che genera correttamente (anche per il type checker) i valori: `Persona Zeus` che ha come figli `Ares` e `Atena`, dove `Ares` e `Atena` possiedono entrambi lo stesso `Oggetto` scudo e in aggiunta hanno rispettivamente un `Oggetto` spada e lancia. Per la creazione dei valori è possibile usare la notazione `Tipo v = { a: ..., ..., z: ... }` e `v.a =`

FOGLIO 3

- ▷ 8. Date le dichiarazioni in Java riportate sotto, qual'è il valore stampato dall'istruzione `print`? Spiegare brevemente il ragionamento seguito — gli schemi delle vtable e dei campi degli oggetti possono aiutare. Determinato il valore stampato per `x = 2`, quale è il valore per `x = 5`?

```

class A {
  int i = 0;
  public A c( A a, int n ) {
    if( n <= 1 ){ i = 1; }
    else {
      int b = new B().c( new A(), a.i() ).i;
      int c = new B().c( new A(), n - 1 - a.i() ).i;
      i = i + b * c;
    }
    return this;
  }
  public int i(){ return i; }
}

class B extends A {
  int i = 0;
  public A c( A a, int n ) {
    a.c( this, n );
    if( ++i < n ){ c( a, n ); }
    return a;
  }
  public int i(){ return i; }
}

```

```

A b = new B();
int x = 2;
print( b.c( new A(), x ).i );

```