

Fondamenti dell'Informatica (completo)

23 gennaio 2017

Esercizio 1

Sia $\Sigma = \{0, 1\}$. Si dimostrino o si confutino le seguenti asserzioni:

- (a) $\forall L_1, L_2 \in \wp(\Sigma^*) : (L_1 + L_2)^* L_2 = (L_1^* L_2)^*$;
- (b) $\forall L_1, L_2 \in \wp(\Sigma^*) : (L_1 L_2 + L_1)^* L_1 = L_1 (L_2 L_1 + L_1)^*$.

Esercizio 2

Si dica se il seguente problema è decidibile o meno, dimostrando formalmente ogni affermazione: dato un linguaggio regolare L qualsiasi su un alfabeto $\Sigma \neq \emptyset$, dire se $\exists k \in \mathbb{N} . L = \Sigma^k$.

Esercizio 3

Si dimostri formalmente che la funzione $f: \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$ definita, per ogni $x, y \in \mathbb{N}$, da

$$f(x, y) = \begin{cases} 1, & \text{se } x + y \geq 1, \\ 0, & \text{altrimenti,} \end{cases}$$

è primitiva ricorsiva. Si utilizzi tale risultato per dimostrare che, se $S, T \subseteq \mathbb{N}$ sono ricorsivi, allora $S \cup T$ è ricorsivo.

Esercizio 4

Si consideri il seguente programma, in un linguaggio con interi a 32 bit ma non meglio identificato:

```
int x = 1;
int w = 3;
int z = 10;

int f(int y) {
    y = 20;
    throw E;
    return (x++) + y;
}

int g(int y) {
    y = 100;
    try { throw E; } catch E {}
    return (x++) + y;
}

try { f(w); } catch E {}
print(x,w,z);
z = g(w);
print(x,w,z);
```

Si dica cosa stampa il programma in ciascuna delle seguenti ipotesi:

1. parametri passati per valore;
2. parametri passati per riferimento;
3. parametri passati per valore-risultato.

Per rispondere, compilare una tabella della forma

1	a_1	b_1	c_1	d_1	e_1	f_1
2	a_2	b_2	c_2	d_2	e_2	f_2
3	a_3	b_3	c_3	d_3	e_3	f_3

Esercizio 5

Si diano schemi di compilazione quanto più generali possibili per il *case statement* alla Pascal e per lo *switch statement* alla C.