

# Fondamenti dell'Informatica

26 giugno 2017

## Esercizio 1

Sia  $I$  un insieme di indici tale che  $|I| \leq \aleph_0$ . Per ogni  $i \in I$ , sia  $S_i \leq \aleph_0$ . Si dimostri che  $|\bigcup_{i \in I} S_i| \leq \aleph_0$ .

## Esercizio 2

Sia collochi il seguente linguaggio nella gerarchia di Chomsky, dimostrando formalmente ogni affermazione:

$$L = \{ b_1 \cdots b_n \in \{0, 1\}^* \mid \forall i = 1, \dots, n : b_i = b_{n-i+1} \}.$$

## Esercizio 3

Sia  $D = \{ x \in \mathbb{N} \mid \varphi_x(x) \uparrow \}$ . Si dimostri che  $D$  non è ricorsivamente enumerabile.

## Esercizio 4

Si consideri il seguente programma:

```
void swap (int a, int b) {  
    int temp;  temp = a;  a = b;  b = temp;  
}  
  
void main () {  
    int value = 2, list[5] = {1, 3, 5, 7, 9};
```

```

    swap(value, list[0]);
    swap(list[0], list[1]);
    swap(value, list[value]);
}

```

Per ognuna delle seguenti modalità di passaggio dei parametri, si dica quali sono i valori delle variabili `value` e `list` dopo ognuna delle tre chiamate a `swap`: per valore, per riferimento, per nome, per valore-risultato. La risposta va fornita compilando quattro tabelle come la seguente:

<i>modalità</i>	<code>value</code>	<code>list[0]</code>	<code>list[1]</code>	<code>list[2]</code>	<code>list[3]</code>	<code>list[4]</code>
dopo 1 <sup>a</sup> <code>swap</code>						
dopo 2 <sup>a</sup> <code>swap</code>						
dopo 3 <sup>a</sup> <code>swap</code>						

dove  $modalità \in \{\text{valore, riferimento, nome, valore-risultato}\}$ .

## Esercizio 5

Si spieghi dettagliatamente come le seguenti dichiarazioni Ada, che fanno uso di record con varianti, potrebbero essere approssimate in C++ o in C. Si scriva il codice C++ o C corrispondente alla tecnica illustrata.

```

1  type POWER is (GAS, STEAM, DIESEL, NONE);
2
3      type VEHICLE (Engine : POWER) is
4          record
5              Model_Year : INTEGER range 1888..1992;
6              Wheels      : INTEGER range 2..18;
7              case Engine is
8                  when GAS      => Cylinders      : INTEGER range 1..16;
9                  when STEAM   => Boiler_Size   : INTEGER range 5..22;
10                     Coal_Burner : BOOLEAN;
11                  when DIESEL => Fuel_Inject   : BOOLEAN;
12                  when NONE    => Speeds        : INTEGER range 1..15;
13              end case;
14          end record;
15
16  Ford      : VEHICLE(GAS);
17  Truck     : VEHICLE(DIESEL);
18  Schwinn   : VEHICLE(NONE);
19  Stanley   : VEHICLE(STEAM);

```