

Problemi ricorrenti e soluzioni paradigmatiche (*goals & plans*)

Violetta Lonati

Università degli studi di Milano
Dipartimento di Informatica

Laboratorio di algoritmi e strutture dati
Corso di laurea in Informatica

Problemi ricorrenti

- ▶ ripetizione
- ▶ conteggio
- ▶ calcolo di un totale (somma, prodotto, ecc.)
- ▶ calcolo del valore estremo (min/max)
- ▶ ricerca di un valore
- ▶ elaborazione su elementi adiacenti

Ripetizione - esempio

Ripetizione di un'azione o sequenza di azioni

Listing 1:

```
for i := 0; i < K; i++ {  
    fmt.Println("*")  
}
```

Conteggio - esempio

Conteggio dei valori di una serie (eventualmente, con una data caratteristica)

Listing 2:

```
var num int
count := 0
for i := 0; i < K; i++ {
    fmt.Scan(&num)
    if num % 3 == 0 {
        count++
    }
}
fmt.Println(count)
```

Calcolo di un totale - esempio

Listing 3:

```
var num int
sum := 0
for i := 0; i < K; i++ {
    fmt.Scan(&num)
    sum = sum + num
}
fmt.Println(sum)
```

Valore estremo - esempio

Listing 4:

```
var n int
fmt.Scan(&n)
max := n
for i := 1; i < K; i++ {
    fmt.Scan(&n)
    if n > max {
        max = n
    }
}
fmt.Println(max)
```

Ricerca lineare - esempio

Identificazione della prima occorrenza di un dato valore (e/o della sua posizione) in una serie

Listing 5:

```
var value, lookedFor int
lookedFor = T
position := -1
for i := 0; i < K; i++ {
    fmt.Scan(&value)
    if value == lookedFor {
        position = i
        break
    }
}
fmt.Println(position)
```

Elaborazione di valori adiacenti - esempio

Listing 6:

```
fmt.Scan(&current)
for i := 1; i < K; i++ {
    previous = current
    fmt.Scan(&current)
    fmt.Println(current - previous)
}
```

