

# Algoritmi & Strutture Dati

Andrea Comar

October 2024

# Contents

<b>I</b>	<b>Algoritmi</b>	<b>3</b>
<b>1</b>	<b>Algoritmi di ordinamento</b>	<b>3</b>
1.1	Insertion sort . . . . .	3
<b>II</b>	<b>Strutture Dati</b>	<b>3</b>
<b>2</b>	<b>strutture dati lineari</b>	<b>3</b>
2.1	array . . . . .	3

## Part I

# Algoritmi

## 1 Algoritmi di ordinamento

**PROBLEMA** Problema data una sequenza  $a_1, a_2, \dots, a_n$  di numeri, trovare una permutazione tale che

$a_1 \leq a_2 \leq \dots \leq a_n$  Soluzioni:

### 1.1 Insertion sort

```
INSERTION-SORT(A)
for (i ← 2 to A.length){
    key ← A[i]
    j ← i - 1
    while (j > 0 && A[j] > key){
        A[j + 1] ← A[j]
        j ← j - 1
    }
}
```

**Complessità Spaziale** :  $\theta(1)$  in richiede unicamente 3 interi (i, j, A.length) per memorizzare i valori.

**Complessità Temporale** :

- nel caso migliore:  $\theta(n)$  vettore già ordinato
- nel caso peggiore:  $\theta(n^2)$  vettore ordinato al contrario

**Correttezza** :

## Part II

# Strutture Dati

## 2 strutture dati lineari

### 2.1 array

struttura dati **statica** (= suo spazio di memoria non varia) di n elementi.