

# Esercizi

## Esercizio 1

Discutere la correttezza del SelectionSort utilizzando l'invariante di ciclo indicata a lezione.

## Esercizio 2

Analizzare il costo in tempo degli algoritmi discussi a lezione utilizzando la notazione asintotica.

## Esercizio 3

Sia  $a$  un array **non ordinato** di dimensione  $n$ , che contiene tutti gli interi compresi tra 1 e  $n + 1$  ad eccezione di uno di essi. Si vuole stabilire l'elemento mancante

Indicare se è possibile o meno risolvere il problema con le seguenti complessità:

- |                     |               |
|---------------------|---------------|
| • tempo $O(n^2)$    | Spazio $O(1)$ |
| • tempo $O(n)$      | Spazio $O(n)$ |
| • tempo $O(n)$      | Spazio $O(1)$ |
| • tempo $O(\log n)$ | Spazio $O(n)$ |

## Esercizio 4

Progettare un algoritmo che verifica se un array  $a$  di  $n$  interi è ordinato in ordine non crescente e analizzarne la complessità (numero di confronti e costo in tempo).

## Esercizio 5

Progettare un algoritmo efficiente per determinare se un array **ordinato** di  $n$  interi, contenente solo 0 e 1, contiene più 0 di 1. Analizzare la complessità dell'algoritmo trovato.

## Esercizio 6

Dati due array  $a$  e  $b$ , di  $n$  e  $m$  interi distinti, progettare un algoritmo per determinare il numero di elementi comuni di  $a$  e  $b$ .

## Esercizio 7

Dati due array **ordinati**  $a$  e  $b$ , di  $n$  e  $m$  interi distinti, progettare un algoritmo per determinare il numero di elementi comuni di  $a$  e  $b$ .

## Esercizio 8

Dato un array **ordinato** di  $n$  interi positivi, progettare un algoritmo efficiente per verificare se esistono due elementi nell'array la cui somma è  $k$ .

## Esercizio 9

Progettare un algoritmo per ordinare **in loco** un array  $a$  di  $n$  interi, il cui valore può essere solo 0 o 1. L'algoritmo deve richiedere tempo lineare nel caso pessimo e può solo scambiare elementi. In particolare, non può usare contatori per mantenere il numero di elementi di un certo valore. Dimostrare la correttezza dell'algoritmo usando la tecnica dell'invariante di ciclo.

## Esercizio 10

Ordinare le seguenti funzioni per tasso di crescita:

$$(\sqrt{2})^{\log_2 n}, \quad n^2, \quad \log^2 n, \quad n \log n, \quad 2^n, \quad 2^{2^n}, \quad 19, \quad \pi^n, \quad 3^{n-2}, \quad (\log n)^n, \quad n^{\log n}$$

## Esercizio 11

Dimostrare che

$$3n^2 - 2n - 1 \in \Theta(n^2) \quad 2^{n+1} \in \Theta(2^n) \quad 2^{2^n} \notin \Theta(2^n)$$