

Esercizi sugli algoritmi di ordinamento — avanzati

Insertion sort, ricerca e ordinamento di struct in C++

Prima di iniziare. Dichiarare `const int N` e suddividere in funzioni. Confrontare il numero di operazioni dei diversi algoritmi.

Esercizio 1 — Insertion sort

Ordinare un vettore in ordine crescente con l'insertion sort.

insertion sort

Prototipo suggerito

```
void insertionSort(int v[], const int N);
```

Esercizio 2 — Ordina e cerca

Ordinare il vettore, poi cercare un valore con la ricerca binaria.

ordinamento + ricerca

Prototipi suggeriti

```
void ordina(int v[], const int N);  
int ricercaBinaria(int v[], const int N, int valore);
```

Esercizio 3 — Conta i confronti

Ordinare con il selection sort restituendo il numero di confronti effettuati.

selection sort

Prototipo suggerito

```
int selectionSortConfronti(int v[], const int N);
```

Esercizio 4 — Ordina per valore assoluto

Ordinare un vettore di interi in base al loro valore assoluto.

ordinamento

Prototipo suggerito

```
void ordinaPerValoreAssoluto(int v[], const int N);
```

Esercizio 5 — Ordina struct

Dato un vettore di studenti (nome e media), ordinarlo per media decrescente.

ordinamento + struct

Struct e prototipo suggerito

```
struct Studente { string nome; float media; };  
void ordinaPerMedia(Studente s[], const int N);
```

Esercizio 6 — Selection decrescente

Ordina un vettore in ordine decrescente con il selection sort.

selection sort

Prototipo suggerito

```
void selectionSortDesc(int v[], const int N);
```

Esercizio 7 — k-esimo più piccolo

Trova il **k**-esimo valore più piccolo di un vettore (ordinando o con altra strategia).

ordinamento

Prototipo suggerito

```
int kesimoPiuPiccolo(int v[], const int N, int k);
```

Esercizio 8 — Fusione ordinata

Fondi due vettori già ordinati in un unico vettore ordinato (come nel merge sort).

merge

Prototipo suggerito

```
void fondi(int a[], const int NA, int b[], const int NB, int ris[]);
```

Esercizio 9 — Ordina stringhe

Ordina alfabeticamente un vettore di stringhe.

ordinamento + string

Prototipo suggerito

```
void ordinaStringhe(string v[], const int N);
```

Esercizio 10 — Ordina con conteggio

Dato un vettore di voti tra 1 e 10, ordinalo usando il conteggio delle occorrenze (counting sort).

counting sort

Prototipo suggerito

```
void countingSort(int v[], const int N, int massimo);
```