

Laboratorio 4 — Rilevamento Duplicati e Compressione

Live Demo — `duplicates_compress.c`

Università di Catania — Dipartimento di Matematica e Informatica

Corso di Laurea in Informatica

Programmazione 1 e Laboratorio (Canale O-Z)

Descrizione del Programma

Si scriva un programma in C, chiamato `duplicates_compress.c`, che legga un array di N interi e lo comprima in-place rimuovendo tutte le occorrenze successive dei valori duplicati.

Il programma deve strutturarsi in funzioni modulari per isolare le operazioni di calcolo e modifica dell'array tramite puntatori.

Requisiti di Implementazione

PUNTO A: Leggere da standard input la dimensione N dell'array (massimo 100000) e successivamente i relativi elementi interi.

PUNTO B: Scrivere la funzione:

```
void trova_estremi(const int *arr, int n,  
                  int *min_idx, int *max_idx);
```

che scandisca l'array e restituisca, mediante i puntatori di output, gli indici del minimo e massimo valore.

PUNTO C: Scrivere la funzione:

```
int conta_occorrenze(const int *start,  
                    int length, int valore);
```

che conti quante volte compare `valore` nel sotto-segmento di array che parte da `start` per una lunghezza pari a `length`.

PUNTO D: Scrivere la funzione di compressione in-place:

```
void compatta_array(int *arr, int *n);
```

che scansioni l'array e rimuova tutti i duplicati traslando a sinistra gli elementi rimanenti e decrementando la dimensione dell'array nel chiamante tramite il puntatore `n`.

PUNTO E: Nel `main`, richiamare la funzione `trova_estremi` per stampare minimo e massimo iniziali con i rispettivi indici. Successivamente, invocare `compatta_array` e stampare l'array compattato e la sua nuova dimensione.

Output di Controllo (Esempi di Esecuzione)

Caso 1 — Esempio generico con duplicati

```
Inserisci la dimensione dell'array: 6
Inserisci gli elementi dell'array:
5 10 5 8 10 12
Estremi: Minimo = 5 (indice 0), Massimo = 12 (indice 5)
Array compattato (nuova dimensione 4, originali 6):
5 10 8 12
```

Caso 2 — Tutti elementi uguali

```
Inserisci la dimensione dell'array: 5
Inserisci gli elementi dell'array:
2 2 2 2 2
Estremi: Minimo = 2 (indice 0), Massimo = 2 (indice 0)
Array compattato (nuova dimensione 1, originali 5):
2
```