

Laboratorio di Informatica

Corso di laurea triennale in Fisica

CARLO MEREGHETTI

In questa esercitazione approfondiremo ulteriormente i seguenti strumenti:

- lettera e scrittura da/su file,
- utilizzo di array dinamici,
- utilizzo di `struct` per modellare dati,
- modularizzazione del codice mediante funzioni in librerie,
- utilizzo di compilazione separata e `makefile`.

Tutti i file (sorgenti ed eseguibili) prodotti andranno salvati sotto la directory `Lab8`.

CERCHI NEL GRANO (...PIANO)

Il file `codici.in`, che si trova nella cartella `Lab7`, contiene un numero non precisato di *numeri interi*. Ciascun numero intero rappresenta un *codice*. Il file `cerchi.in`, nella stessa cartella, contiene invece un numero imprecisato di *cerchi* in \mathbf{R}^2 . Ciascun cerchio è caratterizzato da una quadrupla (x, y, r, c) dove x e y (di tipo `float`) sono le coordinate del centro del cerchio (rispetto ad un sistema di riferimento assegnato), r (`float`) il raggio del cerchio e c (`int`) è il codice che descrive la “specie” del cerchio.

- Progettate il tipo `cerchio` con cui definire variabili che possano contenere tali cerchi.

Scrivere un programma che svolga le seguenti operazioni:

1. Contare e caricare in un array (di interi) i codici presenti nel file `codici.in`. Stampare a video l'elenco dei codici.
2. Contare e caricare in un array (di tipo base `cerchio`) i cerchi descritti nel file `cerchi.in`.
3. Eliminare (*scremare*) tutti i cerchi il cui codice non appartenga all'insieme dei codici caricati al punto 1 e stampare a video il numero di cerchi eliminati e il numero di cerchi rimasti.
4. Per ogni codice individuato al punto 1, e *usando solo i cerchi rimasti dopo la scrematura al punto precedente*:
 - (a) Stampare a video il codice. Determinare il numero di cerchi con quel codice.
 - (b) Calcolare e stampare a video la media delle coordinate x e y dei centri dei cerchi con quel codice, ovvero calcolare il *baricentro* dei centri dei cerchi dello stesso codice.
 - (c) Stampare a video e sul file `cerchi.out` i cerchi con quel codice esibendo, per ogni cerchio, sia la descrizione sia la distanza del centro dal baricentro individuato al punto 4(b).

Tutte le stampe a video e su file vanno accompagnate dalle relative didascalie (dite cosa state stampando).

Per modularizzare la struttura del codice prevedete la scrittura delle seguenti funzioni in un'apposita libreria `cerchi.C` con file d'intestazione `cerchi.h`:

- `int numeriNelFile(char f[])`: restituisce il numero di numeri presenti nel file di nome `f`; aprite e chiudete il file `f` all'interno di questa funzione. Ovviamente, il numero di codici nel file `codici.in` si otterrà chiamando `numeriNelFile("codici.in")` mentre il numero di cerchi nel file `cerchi.in` si otterrà *dividendo per 4 il risultato restituito dalla chiamata* `numeriNelFile("cerchi.in")`. Fate in modo che tale funzione restituisca `-1` nel caso in cui il file `f` non esiste ed utilizzate tale codice nel `main` per interrompere l'esecuzione del programma.
- `void caricaCodici(int *X, int dim)`: carica i codici contenuti nel file `codici.in` nell'array `X` di dimensione `dim`; aprite e chiudete il file all'interno di questa funzione.
- `void caricaCerchi(cerchio *X, int dim)`: carica i cerchi contenuti nel file `cerchi.in` nell'array `X` di dimensione `dim`; aprite e chiudete il file all'interno di questa funzione.
- `bool esisteCodice(cerchio c, int *X, int dim)`: restituisce `true` se il codice del cerchio `c` compare nell'array di codici `X` di dimensione `dim`, `false` altrimenti.
- `int screma(cerchio *X, int dim, int *C, int dimC)`: screma dall'array di cerchi `X` di dimensione `dim` i cerchi il cui codice non appare nell'array di codici `C` di dimensione `dimC`; la funzione restituisce la posizione in `X` dell'ultimo cerchio avente codice in `C`, altrimenti restituisce `-1`.
- `float dist(float x1, float y1, float x2, float y2)`: calcola la distanza tra i punti (x_1, y_1) e (x_2, y_2) .
- `void stampa(cerchio c)`: stampa a video la descrizione del cerchio `c` nella forma:

Centro in (1.3, -2.3), Raggio 7, Codice 1

- `void stampaSuFile(ofstream &out, cerchio c, float d)`: scrive sul file `out` la descrizione del cerchio `c` seguita dalla sua distanza `d` dal baricentro.

Nota: è sempre meglio passare *per riferimento* alle funzioni gli stream di input o output.

Per un'agile gestione del progetto, è richiesta la scrittura di un `makefile` che consenta la compilazione mediante il comando `make compila` e l'esecuzione mediante il comando `make esegui`.