



Università degli studi di Catania

Corso di Laurea in Fisica - Primo livello - A.A. 2018-2019

Esame di informatica - 17 settembre 2019

Prof. Marco Russo

Questo compito è dedicato alla trasformata di Hough. Tale tecnica, data una qualsiasi curva parametrica, rende possibile effettuare un fitting di dati sperimentali. L'unica condizione è che il numero di parametri della curva fittante sia piuttosto ridotto dato che la procedura è piuttosto complessa dal punto di vista computazionale.

Nel compito noi la adopereremo per individuare la migliore retta che approssima dei punti su un piano.

A tale scopo bisogna scrivere un programma in C che generi un certo numero casuale di punti rumorosi sul piano appartenenti ad una retta nota a priori e, successivamente, a partire proprio da questi punti, trovi numericamente la migliore retta fittante (che dovrebbe essere all'incirca quella di partenza).

Più precisamente il programma deve chiedere da tastiera nell'ordine il numero intero n_p di punti, i valori double di: ϵ , rumore r , coefficiente angolare m , e termine noto q . Deve poi scriverli tutti su schermo e chiedere conferma se i valori acquisiti siano effettivamente quelli desiderati. La conferma deve derivare dalla immissione di una qualsiasi stringa che inizi con uno qualsiasi dei caratteri appartenenti alla stringa "sSyY". In tutti gli altri casi il programma deve ricominciare chiedendo nuovamente i valori.

Successivamente, occorre salvare in memoria n_p punti generati casualmente. Per ognuno di questi punti genereremo un valore casuale x_i uniformemente distribuito in $[0, 100]$. Dato ogni valore x_i calcoleremo il corrispondente valore di ordinata della retta indicata da tastiera aggiungendogli un rumore casuale e uniforme in $[-r, r]$.

La seconda parte del programma consiste nel individuare numericamente il coefficiente angolare in $[0, 10]$ e termine noto sempre in $[0, 10]$ che meglio approssimano i punti generati in precedenza. A tal uopo occorre generare esattamente 1000 coefficienti angolari m'_j uniformemente distribuiti e per ognuno di essi altrettanti termini noti q'_k sempre uniformemente distribuiti. Per ogni coppia (x_i, y_i) andare a vedere quanti tra i punti generati in precedenza hanno $|y_i - y'_i| < \epsilon$ (presenza di un hit). Laddove $y'_i = m'_j * x_i + q'_k$. La coppia di valori (m'_j, q'_k) per la quale abbiamo il numero maggiore di hit sarà il fitting desiderato.

Il programma dovrà fornire in output la coppia di valori con 5 cifre dopo la virgola seguito dal numero di hit.

Tutti i calcoli reali dovranno essere effettuati in doppia precisione.

E' vietato l'utilizzo di array statici.

Ci aspettiamo su video qualcosa del tipo:

```
Numero di punti: 1000
Epsilon: 0.001
Rumore: 0.1
m: 1
q: 3
Acquisiti np=1000 eps=0.00 r=0.10 m=1.00 q=3.00. Confermi (s,S,y,Y/altro==NO)?s
Fit migliore m=1.00100 q=2.95295 hits=15
```

oppure

```
Numero di punti: 100
Epsilon: 0.01
Rumore: 0.2
m: 2
q: 4
Acquisiti np=100 eps=0.01 r=0.20 m=2.00 q=4.00. Confermi (s,S,y,Y/altro==NO)?a
Numero di punti: 100
Epsilon: 0.01
Rumore: 0.2
m: 2
q: 4
Acquisiti np=100 eps=0.01 r=0.20 m=2.00 q=4.00. Confermi (s,S,y,Y/altro==NO)?Y
Fit migliore m=1.99199 q=4.57457 hits=8
```

Valutazione del compito.

3 punti	Per l'acquisizione dei valori richiesti da tastiera
5 punti	Per la corretta gestione della conferma
4 punti	Per creazione dell'array dei punti con verifica di memoria sufficiente
5 punti	Per la generazione delle ascisse corrette
5 punti	Per la generazione delle ordinate corrette
5 punti	Per la corretta generazione di tutte le coppie m', q'
8 punti	Per la parte rimanente