

## Ottimizzatore – Esercizio Atleti

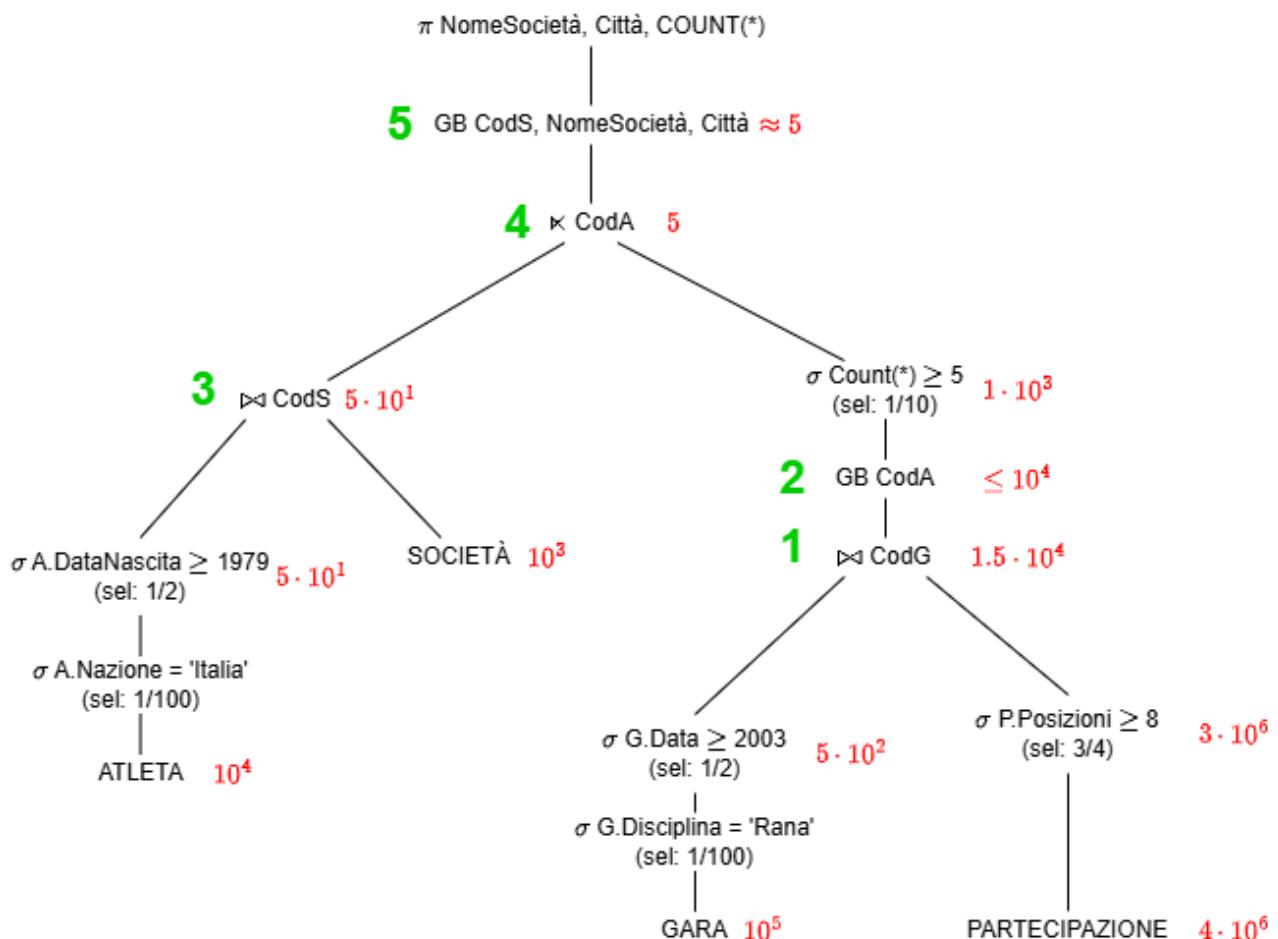
```

SELECT NomeSocietà, Città, COUNT(*)
FROM SOCIETÀ_SPORTIVA S, ATLETA A
WHERE S.CodS = A.CodS and A.Nazione = 'Italia'
and A.DataNascita > 1979 and A.CodA IN
  (SELECT P.CodA
  FROM GARA G, PARTECIPAZIONE P
  WHERE G.CodG = P.CodG and G.Disciplina = 'Rana'
    and G.Data ≥ 2003 and P.Posizione ≥ 8
  GROUP BY P.CodA
  HAVING COUNT(*) ≥ 5 )
GROUP BY S.CodS, NomeSocietà, Città;

```

### Albero algebrico

- $\sigma A.Nazione = 'Italia' \rightarrow =1/100$
- $\sigma A.DataNascita > 1979 \rightarrow \approx 1/2$
- $\sigma G.Data \geq 2003 \rightarrow =1/2$
- $\sigma G.Disciplina = 'Rana' \rightarrow =1/100$
- $\sigma COUNT(*) \geq 5 \rightarrow =1/10$
- $\sigma P.Posizione \geq 8 \rightarrow \approx 3/4$



## Note cardinalità

1. Dopo i filtri, le gare selezionate sono  $5*10^2$ . Assumendo distribuzione uniforme, ci aspettiamo che la stessa percentuale di righe della tabella PARTECIPAZIONE faccia riferimento a queste gare. Applicando quindi il filtro di 3/4 otteniamo quel risultato.
2. Sempre nell'ottica dell'uniformità, non ci aspettiamo che i filtri precedenti abbiano ridotto il numero di atleti coinvolti, che rimangono quindi la totalità di quelli presenti.
3. C'è un rapporto molti:1 tra società e Atleti, perciò la cardinalità è quella degli atleti filtrati.
4. Riduciamo gli atleti di 1/10, riflettendo il filtro del ramo di destra.
5. Non sono stati applicati filtri alle Società, per cui possiamo assumere che la cardinalità rimanga invariata, come caso limite nel quale tutti gli atleti appartengono a società diverse.

## Anticipo GROUP BY

- La GROUP BY su CodA non può essere anticipata in quanto il filtro sul COUNT non restituirebbe lo stesso risultato.
- La GROUP BY su {CodS, NomeSocietà, Città} non può essere anticipata, in quanto facendolo cambierebbe il risultato del COUNT finale. Ad ogni modo, non fornirebbe un vantaggio in termini di riduzione della cardinalità.

## Strutture accessorie

- Tabella "ATLETA" ( $10^4$ )
  - attributo di selezione "Nazione" (1/100)
  - **Ipotesi:**  
`CREATE INDEX IndexA ON ATLETA(Nazione)`
- Tabella "GARA"( $10^5$ )
  - Attributo di selezione "Disciplina" (1/100)
  - CodG potrebbe aiutare il Join
  - Aggiungendo Data, avremmo un indice coprente, che aiuta anche i filtri
  - **Ipotesi:**  
`CREATE INDEX G_idx ON GARA(Disciplina, Data, CodG)`

## Piano di Esecuzione

- Blocco interno (GARA, PARTECIPAZIONE, GROUP BY CodA)  $\rightarrow$  T
  - INDEX RANGE SCAN G\_idx (non è necessaria la lettura della tabella)
  - TABLE ACCESS FULL PARTECIPAZIONE
    - Filtro P.Posizione  $\geq 8$
  - HASH JOIN tra GARA e PARTECIPAZIONE
  - HASH GROUP BY su P.CodA
  - FILTER (HAVING COUNT(\*)  $\geq 5$ )
- Blocco esterno (SOCIETÀ, ATLETA, semi-join con T, GROUP BY finale)
  - INDEX RANGE SCAN A\_idx
  - TABLE ACCESS BY INDEX ROWID ATLETA
  - TABLE ACCESS FULL SOCIETÀ
  - JOIN Nested Loop ATLETA – SOCIETÀ
  - HASH JOIN SEMI ATLETA – T
  - SORT GROUP BY (S.CodS, NomeSocietà, Città)