

Progettazione - Parte A

1. Sono date le relazioni seguenti (le chiavi primarie sono sottolineate):

PUNTO-TAPPA(CodTappa, NomeT, Quota, Regione)

SENTIERO(CodS, NomeS, CodTappaPartenza, CodTappaArrivo, Lunghezza, Dislivello)

RIFUGIO(CodR, NomeR, NumPostiLetto, CodT)

Esprimere le seguenti interrogazioni

- (b) *Obbligatorio* in linguaggio SQL (5 punti): Visualizzare il codice e il nome dei sentieri con dislivello maggiore di 1200 metri per cui non ci sono rifugi né nel punto tappa di partenza né nel punto tappa di arrivo.
- (c) *Opzionale* in linguaggio SQL (5 punti): Per ogni regione, visualizzare codice, nome, quota e regione delle tappe che sono punto di arrivo del numero massimo di sentieri.

```
b) SELECT CodS, NomeS
FROM SENTIERO
WHERE Dislivello > 1200
AND CodTappa Partenza NOT IN
  (SELECT CodT
   FROM RIFUGIO)
AND CodTappa Arrivo NOT IN
  (SELECT CodT
   FROM RIFUGIO)
```

c) SELECT P.CodTappa, Regione, Quota, NomeT
 FROM PUNTO-TAPPA P, SENTIERO S
 WHERE P.CodTappa = S.CodTappaArrivo
 GROUP BY P.CodTappa, Regione, Quota, NomeT
 HAVING COUNT(*) =

(SELECT MAX (TotArrivi)
 FROM PUNTO-TAPPA P1,
 (SELECT CodTappaArrivo,
 COUNT (*) AS TotArrivi
 FROM SENTIERO
 GROUP BY CodTappaArrivo) AS T
 WHERE P1.CodTappa = T.CodTappaArrivo
 AND P1.Regione = P.Regione)

Table function per il calcolo del num. sentieri che arrivano in ogni tappa

calcola il massimo num. di sentieri in arrivo in una tappa per una specificata regione

la regione è definita dalle condizioni di correlazione