

PRODOTTI(Codice, Descrizione, Prezzo_unit)
 MAGAZZINI(Codice, Indirizzo, Città)
 SCORTE(Codice-P, Codice-M, N-pezzi)

1a) Per ogni prodotto che costa più di 100 euro e del quale sono presenti meno di 50 pezzi in almeno un magazzino, si desidera conoscere: i dati del prodotto e l'indirizzo dei magazzini in cui sono presenti meno di 50 pezzi del prodotto.

1b) Per ogni prodotto trovare il magazzino di Roma in cui abbiamo le scorte maggiori (restituire Codice prodotto, Descrizione, Codice magazzino, Indirizzo, N-pezzi)

$$1a) \text{TOT} = (\text{SCORTE} \bowtie_{\text{CODICE}::\text{CODICE-P} \text{ PRODOTTI}} \bowtie_{\text{CODICE}::\text{CODICE-M} \text{ MAGAZZINI}} \text{PI00} = \sigma_{\text{PREZZO_UNIT} > 100 \wedge \text{N-PEZZI} < 50} (\text{TOT}))$$

Query Finale: $Q = \pi_{\text{CODICE-P, DESCRIZIONE, PREZZO-UNIT, INDIRIZZO}} (\text{PI00})$

$$1b) \text{MROMA} = \sigma_{\text{CITTA}::\text{"ROMA"}} (\text{MAGAZZINI}) \quad \text{MagScor} = \text{MROMA} \bowtie_{\text{CODICE}::\text{CODICE-M}} \text{SCORTE}$$

$$\text{MagCopy} = \pi_{\text{PC, MC, NC}} (\rho_{\text{CONES-P} \rightarrow \text{PC, CODICE-M} \rightarrow \text{MC, N-PEZZI} \rightarrow \text{NC}} (\text{MagScor})) \quad \text{COMP} = \sigma_{\text{CP}::\text{CODICE-P} \wedge \text{N-PEZZI} < 50} (\text{MagScor} \times \text{MagCopy})$$

$$P = \text{MagScor} - \pi_{\text{CODICE-P, INDIRIZZO, CODICE-M, N-PEZZI, CITTA}} (\text{COMP}) \quad Q' = \pi_{\text{PRODOTTI, CODICE}::\text{CODICE-P}} (\text{PRODOTTI})$$

Query finale: $Q = \pi_{\text{CODICE-P, INDIRIZZO, CODICE-M, N-PEZZI, DESCRIZIONE}} (Q')$

2) Dato il seguente schema R = ABCDEH sul quale è definito il seguente insieme di dipendenze funzionali:

F = {A->C, AB->D, ABH->CE, BH->AD, D->C, DE->H, E->B}

2a) trovare le chiavi dello schema motivando la risposta*

2b) dire se lo schema è in 3NF motivando la risposta*

2c) trovare una decomposizione dello schema*, in modo tale che ogni sottoschema sia in 3NF, e che la decomposizione preservi F e abbia un join senza perdita.

2a) Noto che C non compare mai come determinante, quindi non farà parte della chiave.

Inizio con i sottoinsiemi di ABDEH.

$ABDE^+_F = R$. Controllo $ABD^+_F = R$. $BDE^+_F = R$, $BD^+_F = R$, $DE^+_F = R$ $BE^+_F = R \Rightarrow DE$ e' chiave.

controllo $ADE \Rightarrow EA^+_F = R \Rightarrow EA$ e' chiave. Controllo $BDEH^+_F \Rightarrow HD^+_F = R$, $EH^+_F = R \Rightarrow EH$ e' chiave.

Controllo $ADEH \Rightarrow AEH^+_F = R \Rightarrow AH^+_F = R$. $ABEH^+_F \Rightarrow AB^+_F = R$, $EB^+_F = R$. $ABH^+_F = R \Rightarrow BH^+_F = R \Rightarrow BH$ e' chiave.

2b) OSS. C non e' primo, A non e' chiave (quindi non e' superchiave) $\Rightarrow A \rightarrow C$ viola la 3NF.

2c) Inizio minimizzando i determinati: $F = \{A \rightarrow C, AB \rightarrow D, \text{ABH} \rightarrow C, ABH \rightarrow E, BH \rightarrow A, BH \rightarrow D, D \rightarrow C, DE \rightarrow H, E \rightarrow B\}$, ora

controllo i determinati: $\bullet AB \rightarrow D, D \notin A^+ \wedge D \notin B^+ \bullet ABH \rightarrow C, C \notin A^+ \Rightarrow$ sostituisco. $\bullet ABH \rightarrow E$ non andrà sostituito, perche' E e'

determinato solo da ABH. $\bullet BH \rightarrow A$ analogo. $\bullet BH \rightarrow D, B^+_F = B, H^+_F = H \bullet DE \rightarrow H$ medesimo di $ABH \rightarrow E$. Ora controllo le

ridondanze. $A^+_{F/A-C} = \{A\}$. $AB^+_{F/AB-D} = \{ABC\}$. $ABH^+_{F/ABH-C} = \{ABHCD\}$. $BH^+_{F/BH-A} = \{BHDC\}$. $BH^+_{F/BH-D} \Rightarrow D$ e' di troppo! $D^+_{F/D-C} = D$. $DE^+_{F/DE-H} = \{BDEC\}$

$B \notin E^+_{F/E-B}$. La copertura minimale e': $F = \{A \rightarrow C, AB \rightarrow D, ABH \rightarrow E, BH \rightarrow A, D \rightarrow C, DE \rightarrow H, E \rightarrow B\}$ Applico l'algoritmo.

C'e' un attributo che non appare in F? no \Rightarrow c'e' una dip. in cui compare tutto R? no \Rightarrow Allora la

decomposizione e': $\rho = \{AC, ABD, ABHE, BHA, DC, DEH, EB\}$, la chiave $DE \in DEH \in \rho \Rightarrow$ Ha un Join senza perdita.

3) E' dato un file di 168.600 record. Ogni record occupa 426 byte, di cui 32 per la chiave. Un blocco contiene 4096 byte. Un puntatore a blocco occupa 5 byte. Si utilizza una organizzazione hash con record distribuiti uniformemente tra 250 bucket.

3a) Calcolare l'occupazione della bucket directory e dei bucket

3b) Calcolare il costo medio di una ricerca considerando chiavi di hash univoche

3c) Quanti bucket occorrerebbero per avere un costo medio di ricerca minore o uguale a 20 accessi?

$$3a) B_{dir} = 250 \cdot 5 = 1250 \text{ byte} \Rightarrow \text{e' necessario 1 blocco. } \text{Record} \times \text{Bucket} = \lceil \frac{168.600}{250} \rceil = 675$$

$$\text{Record} \times \text{Blocco} = \lceil \frac{4096 \cdot 5}{426} \rceil = 9 \quad \text{Blocchi} \times \text{Bucket} = \lceil \frac{675}{9} \rceil = 75$$

$$3b) \text{AvgTime} = \lceil \frac{75}{2} \rceil = 38 \text{ accessi.}$$

$$3c) \frac{x}{2} \leq 20 \Rightarrow x \leq 40 \Rightarrow \text{In un bucket ci devono essere } x \leq 40 \text{ blocchi.}$$

$$\frac{x}{9} \leq 40 \Rightarrow x \leq 40 \cdot 9 \Rightarrow x \leq 360 \text{ quindi } \frac{168.600}{x} \leq 360 \Rightarrow \lceil \frac{168.600}{360} \rceil \leq x \Rightarrow x \geq 469 \text{ bucket}$$