

1, (đề b) PROJ (PNO, PNAME, BUDGET, LOC)

SELECT sum(budget) FROM PROJ WHERE LOC = 'Paris'

Ta có vector $use(q_i, A_j) = \begin{cases} 1 & \text{nếu } q_i \text{ tham chiếu đến thuộc tính } A_j \\ 0 & \text{ngược lại} \end{cases}$

Kí hiệu $A_1 = PNO$

$A_2 = PNAME$

$A_3 = BUDGET$

$A_4 = LOC$

$q_1 = \text{SELECT sum}(A_3) \text{ FROM PROJ WHERE } A_4 = \text{'Paris'}$

$\rightarrow \text{vector } use(q_1, A_i) = [0 \ 0 \ 1 \ 1]$
với $i = 1, 2, 3, 4$

1, (Đề 2)

EMP (ENO, ENAME, TITLE, ASG (ENO, PNO, RESP, DUR))

CREATE VIEW EMP_VIEW AS SELECT EMP.ENO, EMP.ENAME,
ASG.PNO, ASG.RESP

FROM EMP, ASG

WHERE EMP.ENO = ASG.ENO AND ASG.DUR > 24

Ta có vector $use(q_i, A_j) = \begin{cases} 1 & \text{nếu } q_i \text{ tham chiếu tới thuộc tính } A_j \\ 0 & \text{ngược lại} \end{cases}$

CREATE VIEW EMP_VIEW AS SELECT A1, A2, A3, A4

FROM EMP, ASG

WHERE A1 = A4 AND A3 > 24

$use(q_i, A_i) = [1 \ 1 \ 0 \ 1 \ 1 \ 1 \ 1]$

với $i = 1, \dots, 7$

Câu 1:

* Để xác định phân tích:

Đề 4:

Câu 1:

$$R = (A_1, A_2, A_3, A_4, A_5, A_6)$$

$$R_1 = (A_1, A_2, A_5, A_6); R_2 = (A_1, A_3, A_4, A_5)$$

$$R_3 = (A_1, A_5)$$

Kiểm tra tính đúng đắn

→ là phân mảnh dọc

(1) xét tính đầy đủ

$$\Omega = \{A_1, A_2, A_3, A_4, A_5, A_6\}$$

Thấy $\forall A \in \Omega$ luôn có $\exists i \in [1, 3]$ sao cho $A \in R_i$.

⇒ thỏa mãn tính đầy đủ

(2) xét tính phục hồi

$$R_1 \bowtie R_2 \bowtie R_3 = \{A_1, A_2, A_3, A_4, A_5, A_6\} \equiv R$$

⇒ thỏa mãn tính phục hồi

(3) xét tính tách biệt

$$R_1 \cap R_2 = \{A_1, A_5\}$$

$$R_1 \cap R_3 = \{A_1, A_5\}$$

$$R_2 \cap R_3 = \{A_1, A_5\}$$

→ chứng minh thuộc tính ở khóa → o tách biệt
(dù chỉ có 1 thuộc tính ở khóa thôi thì cũng b' tách biệt)

Câu 2:

Đề 2

Câu 5

	A_1	A_2	A_3	A_4	A_5		S_1	S_2	S_3	S_4	
q_1	1	0	0	0	1	q_1	0	20	20	5	45
q_2	0	0	1	1	0	q_2	5	0	20	5	30
q_3	0	1	0	1	1	q_3	10	0	5	15	30
q_4	1	0	0	1	0	q_4	0	20	10	0	30

lg

Đính ma trận AA

$$adj(A_i, A_j) = \sum_{k: (u_k, v_k) \in E} u_k(q_i) \cdot v_k(q_j) + \sum_{k: (u_k, v_k) \in E} u_k(q_i) \cdot v_k(q_j)$$

$$adj(A_1, A_2) = \sum_{k=1,2,3,4} \left(\sum_{l=1}^4 acc_k(q_l) \right)$$

$$= acc_2(q_1) + acc_2(q_2) + acc_3(q_1) + acc_4(q_1) + acc_4(q_3) + acc_2(q_1) + acc_3(q_3) + acc_4(q_3)$$

$$= 20 + 20 + 5 + 20 + 10 = 75$$

$$adj(A_1, A_2) = \sum_{k=1,2,3,4} \left(\sum_{l=1}^4 -acc_k(q_l) \right) = 0$$

$$= acc_2(q_1) + acc_2(q_2) + acc_3(q_1) + acc_4(q_1)$$

$$+ acc_2(q_2) + acc_2(q_3) + acc_3(q_1) + acc_4(q_3)$$

$$+ acc_2(q_1) + acc_2(q_4) + acc_3(q_4) + acc_4(q_4)$$

$$= 20 + 20 + 5 + 10 + 5 + 15 + 20 + 10$$

$$= 105$$

$$adj(A_2, A_3) = \sum_{k=1,2,3,4} \left(\sum_{l=1}^4 -acc_k(q_l) \right) = 0$$

$$= acc_2(q_1) + acc_2(q_2) + acc_3(q_2) + acc_4(q_2)$$

$$+ acc_2(q_2) + acc_2(q_3) + acc_3(q_2) + acc_4(q_2)$$

$$+ acc_2(q_4) + acc_2(q_4) + acc_3(q_4) + acc_4(q_4)$$

$$= 105$$

làm tj từ ta có:

$$adj(A_1, A_4) = 30$$

$$adj(A_2, A_5) = 15$$

$$adj(A_4, A_2) = 30$$

$$adj(A_1, A_2) = 0$$

$$adj(A_2, A_4) = 30$$

$$adj(A_2, A_5) = 75$$

$$adj(A_3, A_3) = 30$$

$$adj(A_3, A_4) = 30$$

$$adj(A_3, A_5) = 15$$

$$adj(A_4, A_4) = 90$$

$$adj(A_4, A_5) = 15$$

p1 chia R thành 2 mảnh
 --J1 (những ai là LTV)
 -- R\J1 (không phải LTV)

$Pr'=\{p1\}$ $Pr=\{p2,p3,p4\}$

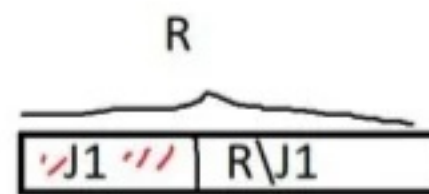
p2 không phân hoạch J1,
 R\J1 thành các mảnh ->
 không TM quy tắc 1

p3: chia J1 thành 2 mảnh
 --J11: LTV, > 3tr ($= \sigma_{p_4}^{-}(J1)$)
 -- J12: LTV, <= 3tr (bằng p4) ($= \sigma_{p_4}(J1)$)

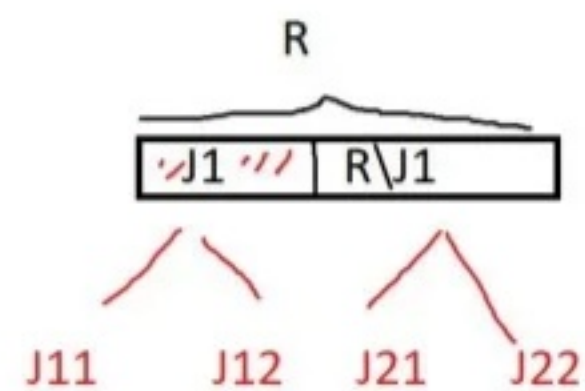
p3 chia R\J1 thành 2 mảnh
 --J21: không phải LTV, > 3tr ($= \sigma_{p_4}(R\setminus J1)$)
 --J22: không phải LTV, <= 3tr ($= \sigma_{p_4}^{-}(R\setminus J1)$)

$Pr'=\{p1, p3\}$ $Pr=\{p2,p4\}$

(Ngầm hiểu p3 làm luôn việc của p4,
 nên kết luận là $Pr'=\{p1,p3\}$ đầy đủ và
 cực tiểu)



p1: nghe= 'LTV'
 p2 : nghe != 'LTV'
 p3 : lương > 3tr
 p4 : lương < 3tr



Date

No.

→ Xếp 4 thư vào túi (A_1, A_2, A_3, A_4)

Xét các cách xếp vào các túi sau:

$(5, 1, 4, 2, 3), (1, 5, 4, 2, 3), (1, 4, 5, 2, 3), (2, 4, 2, 5, 3), (1, 4, 2, 3, 5)$
 $(5, 1, 4, 2, 3), (1, 5, 4, 2, 3), (1, 4, 5, 2, 3), (2, 4, 2, 5, 3), (1, 4, 2, 3, 5)$

$$\begin{aligned} \text{Cost}(A_1, A_2, A_3) &= 2 \text{bond}(A_1, A_2) + 2 \text{bond}(A_2, A_3) - 2 \text{bond}(A_1, A_3) = 13620 \\ \text{Cost}(A_1, A_2, A_4) &= 2 \text{bond}(A_1, A_2) + 2 \text{bond}(A_2, A_4) - 2 \text{bond}(A_1, A_4) = 15920 \\ \text{Cost}(A_1, A_3, A_4) &= 2 \text{bond}(A_1, A_3) + 2 \text{bond}(A_3, A_4) - 2 \text{bond}(A_1, A_4) = 15500 \\ \text{Cost}(A_2, A_3, A_4) &= 2 \text{bond}(A_2, A_3) + 2 \text{bond}(A_3, A_4) - 2 \text{bond}(A_2, A_4) = 8200 \\ \text{Cost}(A_1, A_2, A_3, A_4) &= 2 \text{bond}(A_1, A_2) + 2 \text{bond}(A_2, A_3) + 2 \text{bond}(A_3, A_4) - 2 \text{bond}(A_1, A_3) - 2 \text{bond}(A_1, A_4) - 2 \text{bond}(A_2, A_4) = 1800 \end{aligned}$$

→ Xếp 4 thư vào túi $(A_1, A_2, A_3, A_4, A_5)$

Có ma trận CA:

	A_1	A_2	A_3	A_4	A_5
A_1	95	45	30	0	0
A_2	45	35	30	30	0
A_3	30	30	90	30	30
A_4	0	30	30	30	0
A_5	0	0	30	0	30

Phân nhóm thư:

Tại vị trí 1:

$TA = \{A_1\}$

$BA = \{A_2, A_3, A_4, A_5\}$

→ $TQ = \{95\}$

$BQ = \{92, 93\}$

$OQ = \{91, 94\}$

→ $CTQ = 60$

$CBQ = 60$

$COQ = 75$

$$\rightarrow z = CTQ \cdot CBQ - COQ^2 = -75 = -75$$

Tại vị trí 2:

$TA = \{A_2, A_3\}$

$BA = \{A_1, A_4, A_5\}$

→ $TQ = \{92\}$

$BQ = \{92, 93\}$

$OQ = \{93, 94\}$

→ $CTQ = 45$

$CBQ = 30$

$COQ = 60$

$$\rightarrow z = 45 \cdot 30 - 60^2 = -2250$$

Date

Tại vị trí 3:

$TA = \{A_3\}$

$BA = \{A_1, A_2, A_4, A_5\}$

Tại vị trí 4:

$TA = \{A_4\}$

$BA = \{A_1, A_2, A_3, A_5\}$

→ A

ĐỀ 9:

Phân nhóm:

A_1, A_2, A_3

$q_1 = 1, 1$

$q_2 = 1, 1$

$q_3 = 1, 0$

$q_4 = 0, 0$

Tìm ma

$q_{ij}(A_1)$

$q_{ij}(A_2)$

$q_{ij}(A_3)$

$q_{ij}(A_4)$

$q_{ij}(A_5)$

$q_{ij}(A_6)$

$q_{ij}(A_7)$

$q_{ij}(A_8)$

$q_{ij}(A_9)$

$q_{ij}(A_{10})$

$q_{ij}(A_{11})$

$q_{ij}(A_{12})$

HONG HA

1, (đề b) PROJ (PNO, PNAME, BUDGET, LOC)

SELECT sum(budget) FROM PROJ WHERE LOC = 'Paris'

Ta có vector $use(q_i, A_j) = \begin{cases} 1 & \text{nếu } q_i \text{ tham chiếu đến thuộc tính } A_j \\ 0 & \text{ngược lại} \end{cases}$

Kí hiệu $A_1 = PNO$

$A_2 = PNAME$

$A_3 = BUDGET$

$A_4 = LOC$

$q_1 = \text{SELECT sum}(A_3) \text{ FROM PROJ WHERE } A_4 = \text{'Paris'}$

$\rightarrow \text{vector } use(q_1, A_i) = [0 \ 0 \ 1 \ 1]$
với $i = 1, 2, 3, 4$

1, (đề 2)

EMP (ENO, ENAME, TITLE, ASG (ENO, PNO, RESP, DUR))

CREATE VIEW EMP_VIEW AS SELECT EMP.ENO, EMP.ENAME,
ASG.PNO, ASG.RESP

FROM EMP, ASG

WHERE EMP.ENO = ASG.ENO AND ASG.DUR > 24

Ta có vector $use(q_i, A_j) = \begin{cases} 1 & \text{nếu } q_i \text{ tham chiếu tới thuộc tính } A_j \\ 0 & \text{ngược lại} \end{cases}$

CREATE VIEW EMP_VIEW AS SELECT A1, A2, A3, A4

FROM EMP, ASG

WHERE A1 = A4 AND A3 > 24

$use(q_i, A_i) = [1 \ 1 \ 0 \ 1 \ 1 \ 1 \ 1]$

với $i = 1, \dots, 7$

CSDLPT

LT

1, CSDLPT là gì? Các kiểu PM, điều kiện PM, thông tin PM?

- Là 1 tập các CSDL có QH với nhau và một logic và được phân bố trên 1 mạng máy tính, thể hiện với 1 dùng / ứng dụng như 1 hệ thống đơn lẻ

- Phần lớn CSDL được thiết kế theo kiểu top-down. Thiết kế phân mảnh là việc đầu tiên phải thực hiện. Mục đích phân mảnh là tạo ra các đơn vị cấp phát logic sao cho chi phí thực hiện truy vấn là thấp nhất.

- Các kiểu PM:

+ Dọc: Các quan hệ được chia theo chiều dọc. Các QH mảnh có thể có 1 số thuộc tính chung (khóa) khi kết nối tự nhiên các mảnh sẽ khôi phục QH gốc. \Rightarrow Phép diễn lên tập con thuộc tính QH gốc.

- Ngang: 1 quan hệ đc chia theo điều ngang thành nhiều mảnh khác nhau. Các mảnh là các quan hệ khả hợp với quan hệ gốc.
 \Rightarrow Phép chia quan hệ gốc thỏa mãn điều kiện.

* Các mảnh ko có bộ chung.

- ĐK phân mảnh: Tính đúng đắn.

+ Tính đầy đủ:

Ngang: $\forall x \in R(\Omega) \Rightarrow \exists i \in [1..k] / x \in R_i(\Omega)$

Doc: $\forall A \in \Omega \Rightarrow \exists i \in [1..k] / A \in \Omega_i$

+ Tính tái tạo (phục hồi):

Ngang: $R = R_1 \cup R_2 \cup \dots \cup R_k$

Doc: $R = R_1 \bowtie R_2 \bowtie \dots \bowtie R_k$

+ Tính tách biệt:

Ngang: $\forall i \neq j \in [1..k] \Rightarrow R_i \cap R_j = \emptyset$

Doc: $\forall i \neq j \in [1..k] \Rightarrow \Omega_i \cap \Omega_j = \{\pi \text{ khóa}\}$

- Thông tin phân mảnh :

+ Thông tin về 'CSDL' bao gồm tập các quan hệ, tập các phụ thuộc.

+ Thông tin các ứng dụng bao gồm các câu truy vấn trên quan hệ.

+ Thông tin về 'mạng máy tính, cấu trúc, băng thông...

+ Thông tin về 'hệ thống máy tính, bộ nhớ lưu trữ...

(Chỉ SD trong mô hình cập phát, không sử dụng trong các thuật toán phân mảnh dữ liệu).

2, KN phân mảnh, phân mảnh ngang, dọc ? Thông tin và thuật toán.

- Phân mảnh là phân rã 1 quan hệ thành nhiều mảnh, mỗi mảnh được xử lý như 1 đơn vị dữ liệu, cho phép thực hiện nhiều giao dịch đồng thời hay thực hiện song song các vấn đề làm tăng mức độ hoạt động đồng thời và lưu lượng hoạt động của hệ thống.

- PM ngang: (T^2 cân thăng)

+ PM ngang nguyên thủy: là PM ngang đã thực hiện trên các vị trí của đỉnh ghế đồ.

+ PM ngang dẫn xuất: là phân số 1 quan hệ dựa trên các vị trí của quan hệ \neq .

Nhưng: $R_i = G_{F_i}(R) \quad 1 \leq i \leq Z$

Dẫn xuất: $R_i = R \times S_i \quad 1 \leq i \leq W$

($S_i = G_{F_i}(S)$)

* F_i là công thức định nghĩa PM ngang nguyên thủy.

- Thông tin PM:

+ CSDL: là các thông tin về 'lửa đồ' khái niệm toàn cục của hệ CSDL.

+ Ứng dụng: Để thực hiện phân mảnh cần có thông tin định tính và định lượng. TTĐT hướng dẫn cho hoạt động phân mảnh, TTĐL sử dụng cho mô hình cập phát.

COM MIN

Vào R: Quan hệ.

P_x : Tập các vị trí đồng giá.

P_x, P_x' : Tập các vị trí đồng giá.

Khai báo: F : Tập mã lời sơ cấp.

Begin: Tìm 1 vị trí $p_i \in P_x$ / p_i phân hoạch R theo quy tắc.

$P_x' \leftarrow p_i$, $P_x \leftarrow P_x - p_i$, $F \leftarrow f_i$ { f_i là mã lời sơ cấp theo p_i }

do

begin

Tìm 1 $p_j \in P_x$ / p_j phân hoạch 1 mã lời của P_x' theo quy tắc

$P_x' \leftarrow P_x' \cup p_j$ | $P_x \leftarrow P_x - p_j$ | $F \leftarrow F \cup f_j$

If $\exists p_k \in P_x'$ không liên đới then

begin $P_x' \leftarrow P_x' - p_k$

$F \leftarrow F - p_k$

end if

end begin

until P_x đầy đủ

End