2

BÀI GIẢI

Bài giải Chương 1

QUAN HỆ VÀ ĐẠI SỐ QUAN HỆ

Cơ sở dữ liệu minh họa: CSDL Thực tập

SV(SV#, HT, NS, QUE, HL)

DT(*DT#, TDT, CN, KP*)

SD(SV#, DT#, NTT, KM, KQ)

***SV* - tên quan hệ sinh viên**

SV# - mã số sinh viên

HT - họ và tên sinh viên

NS - năm sinh của sinh viên

QUE - quê (tỉnh)

HL - học lực thể hiện qua điểm trung bình

***DT* - tên quan hệ đề tài**

DT# - mã số đề tài

TDT - tên đề tài

CN - họ và tên chủ nhiệm đề tài

KP - kinh phí cấp cho đề tài (triệu đồng).

***SD* - tên quan hệ sinh viên - đề tài**

SV# - mã số sinh viên

DT# - mã số đề tài mà sinh viên đó tham gia

NTT - nơi thực tập để triển khai đề tài (tỉnh)

KM - khoảng cách từ nơi thực tập đến trường

KQ - kết quả thực tập theo đề tài đã chọn.

Giả thiết là một sinh viên có thể tham gia nhiều đề tài, mỗi đề tài sinh viên đó thực tập tại một địa điểm.

1.1.

```
a) R[AB]=(A B)
          a 1
          b 2
          a 4
          c 5
```

```
b) R(3-B+D>1) =
   (A B C D)
    a 1 x 2
    a 1 y 2
    b 2 x 1
    b 2 y 1
    c 5 y 7
```

```
c) R(B<4)+R(D>3) =
   (A B C D)
    a 1 x 2
    a 1 y 2
    b 2 x 1
    b 2 y 1
    c 5 y 7
```

```
d) R(B>=1&B<=5)=
   (A B C D)      = R
    a 1 x 2
    a 1 y 2
    b 2 x 1
    b 2 y 1
    a 4 x 2
    c 5 y 7
```

```
e) R*S[C]
=(A B C D) = R
  a 1 x 2
  a 1 y 2
  b 2 x 1
  b 2 y 1
  a 4 x 2
  c 5 y 7
```

```
f) R(B<4)-R(D>3)
= (A B C D)
   a 1 x 2
   a 1 y 2
   b 2 x 1
   b 2 y 1
```

```
h) R:S =(A B)
        = ∅
```

```
g) R(B<4)& R(D>3)
=(A B C D)
= ∅
```

1.2. *SV(NS<1983 & QUE="Hai Phòng")*

1.3. *SD[NTT]*

1.4. *SD(DT#=7 & KM > 100)[NTT]*

1.5. *SD(NTT="Nha Trang")*

***1.6.** *((SV[SV#,HT,QUE])*(SD[SV#,NTT]))(QUE=NTT)[HT]*

1.7. *DT*SD[DT#]*

1.8. *(DT[DT#])-(SD[DT#])*

1.9. *DT(KP=1.5 | KP>2)[DT#]*

1.10. *SV(2003-NS<20)[SV#]*(SD(KQ>7)[SV#])*

***1.11.** *SD[DT#,NTT] : (SD(DT#=1)[NTT])*

***1.12.** *(DT[DT#,TDT]*(SD[DT#,NTT]:SD[NTT]))[TDT]*

***1.13.** *(SV*(SD*(DT(KP*5>Sum(DT,KP))[DT#]))[SV#])[HT]*

***1.14.** *SV(HL>Avg(SD(DT#=4),KQ))[HT]*

1.15. Suy trực tiếp từ định nghĩa

1.16. Thí dụ chứng tỏ các phép toán *trừ* và *chia không có tính giao hoán.*

R = (A)	S (A)
a	a
b	
R-S = (A)	
b	
S-R = ∅	

R = (A B)	S (B)
a 1	1
b 2	
R : S = (A)	
a	
S : R = ∅	

1.17.

a) $R(e \& h) = R(h \& e)$, vì $e\&h = h\&e$.

b) $R(e \& h) = R(e) \& R(h) \subseteq R(e)$, vì $e \& h \Rightarrow e$.

c) $R(e \& h) = R(h) \& R(e) \subseteq R(h)$, vì $e \& h \Rightarrow h$.

d) $R(e \& h) = R(e)(h)$, vì $\forall t \in R$: $Sat(t, e\&h) \Leftrightarrow Sat(t,e) \& Sat(t,h)$.

e) $R(e \mid h) = R(h \mid e)$, vì $\forall t \in R$: $Sat(t, e \mid h) \Leftrightarrow Sat(t,e) \mid Sat(t,h)$.

f) $R(e \mid h) = R(e) + R(h)$, xem e.

g) $R(!\, e) = R - R(e)$, vì $\forall t \in R$: $Sat(t, !\, e) \Leftrightarrow !\, Sat(t,e)$

h) $R(True) = R$, vì $\forall t \in R$: $Sat(t, True)$

i) $R(False) = \varnothing$, vì $\forall t \in R$: $!\, Sat(t, False)$

***1.19.** $\forall t \in R : S \Leftrightarrow t \in R[M] \& t*S \subseteq R \Leftrightarrow \exists u \in R$: $u.M = t \& t*S \subseteq R \Leftrightarrow t=u.M \in R[M] \& u \notin R[M]*S - R \Leftrightarrow t \in R[M] \& t \notin (R[M]*S - R)[M] \Leftrightarrow t \in R[M] - (R[M]*S - R)[M]$.

1.20.

Phép toán	Ký hiệu	Tính đóng
chọn	( )	có
chiếu	[ ]	có
kết nối tự nhiên	*	có
cộng	+	không *
giao	&	không *
trừ	-	không *
chia	:	không

(*) Các phép toán quan hệ cộng, giao, trừ không đóng với hai quan hệ không tương thích.

1.21. Suy trực tiếp từ định nghĩa.

1.22. a) Thí dụ chứng tỏ phép toán trừ không có tính kết hợp:

Cho tập thuộc tính tuỳ ý $U \neq \varnothing$. Chọn 3 bộ khác nhau tuỳ ý trên U là t, u, v. Xây dựng các quan hệ $R = \{ t, u \}$; $S = R$; $T = \{ u, v \}$. Ta có

$R-(S-T) = R-(R-T) = \{u\} \neq (R-S)-T = (R-R)-T = \varnothing - T = \varnothing$.

a) Thí dụ chứng tỏ phép chia không có tính kết hợp:

Xét các quan hệ $R(A,B)$, $S(B,C)$ và $T(C)$. Ta có

$R:(S:T)$ cho ta quan hệ có thuộc tính A, trong khi biểu thức $(R:S):T$ không có nghĩa.

1.26.

Phép toán	Ký hiệu	Nở/Co ngang	Nở/Co dọc
Chọn	( )	=	-
Chiếu	[ ]	-	-
Kết nối tự nhiên	*	+	+ -
Cộng	+	=	+
Giao	&	=	-
Trừ	-	=	-
Chia	:	-	-

***1.29.** Chẳng hạn, R(False) hoặc $R(A \neq A)$ với A là một thuộc tính bất kỳ trong U.

***1.33.** Thí dụ chứng tỏ $R[X_1] * R[X_2] * \ldots * R[X_k] \neq R$

R = (A B C)	P = $R[AB]$	Q = $R[BC]$	$P*Q$ = (A B C) ≠ R
a 1 x	= (A B)	= (B C)	a 1 x
b 1 y	a 1	1 x	a 1 y
	b 1	1 y	b 1 x
			b 1 y

***1.36.** *SV* * *DT* cho ta tích Descartes.

Bài giải Chương 2

CÁC THAO TÁC TRÊN BỘ VÀ QUAN HỆ

2.1.

```
Algorithm  Selection
Format: P = R(e)
Input:        - Quan hệ R(U)
              - Biểu thức chọn e trên U.
Output: - Quan hệ
        P(U) = R(e) = {t∈R|Sat(t,e)}
Method
// Tạo lập quan hệ P với tập thuộc  tính
// của quan hệ R
        Create(P,Attr(R));
        for each tuple t in R
            with Sat(t,e) do
                add t to P ;
        endfor;
        return P;
end Selection;
```

2.2.

```
Algorithm  Projection
Format: P = R[X]
Input:          - Quan hệ R(U)
                - Tập con thuộc tính X của U.
Output: - Quan hệ R[X]={t.X | t∈R}
Method
    Create(P,X);
      for each tuple t in R
           with t.X not_in P do
              add t to P ;
       endif;
      endfor;
      return P;
end   Projection;
```

2.3.

```
Algorithm Join
Format: P = R*S
Input:          - Quan hệ R(U)
                - Quan hệ S(V)
Output: - Quan hệ
 R*S={u*v|u∈R,v∈S,u.M = v.M, M = U ∩ V}
```

```
Method
        X = Attr(R) ∪ Attr(S);
        M = Attr(R) ∩ Attr(S);
    Create(P,X);
        for each tuple u in R do
                for each tuple v in S do
                        if u.M=v.M then
                                add u*v to P ;
                        endif;
                endfor;
        endfor;
        return P;
end Join;
```

2.4.

```
Algorithm Union
Format: P = R+S
Input:      - Quan hệ R(U)
            - Quan hệ S(U)
Output: - Quan hệ R+S={t|t∈R ∨ t∈S}
Method
    Create(P,Attr(R));
        for each tuple u in R do
                add u to P;
```

```
        endfor;
        for each tuple v in S
                with v not_in R do
                        add v to P ;
        endfor;
        return P;
end Union;
```

2.5.

```
Algorithm Intersection
Format: P = R&S
Input:      - Quan hệ R(U)
            - Quan hệ S(U)
Output: - Quan hệ R&S={t|t∈R ∧ t∈S}
Method
    Create(P,Attr(R));
      for each tuple u in R
              with u in S do
                      add u to P;
      endfor;
      return P;
end Intersection;
```

2.6.

```
Algorithm Substraction
Format: P = R-S
Input:        - Quan hệ R(U)
              - Quan hệ S(U)
Output: - Quan hệ R-S={t|t∈R ∧ t∉S}
Method
    Create(P,Attr(R));
      for each tuple u in R
            with u not_in S do
                  add u to P;
      endfor;
      return P;
end  Substraction;
```

2.7.

```
Algorithm Division
    Format: P = R:S
    Input:        - Quan hệ R(U)
                  - Quan hệ S(V)
    Output:
    - Quan hệ R:S={t.M|t∈R,(t.M)*S⊆R,M = U-V}
    Method
          M = Attr(R)-Attr(S);
```

```
        c := Card(S);//số bộ của S
    Create(P,M);
      for each tuple t in R
          with t.M not_in P do
             d:=0;//khởi tạo biến đếm
             for each tuple v in S
                  if (t.M)*v in R then
                        d:=d+1
                  else breakfor;
                  endif;
             endfor;
             if d=c then
                  add t.M to P;
             endif;
      endfor;
      return P;
end  Division;
```

2.8.

```
Algorithm  Selection_Projection
Format: P = R(e,X)
Input:        - Quan hệ R(U)
              - Biểu thức chọn e trên U.
              - Tập con X của U.
```

```
Output: - Quan hệ
P(X) = R(e,X) = {t.X | t∈R, Sat(t,e)}
Method
        Create(P,X);
        for each tuple t in R
            with Sat(t,e) do
                if t.X not_in P then
                    add t.X to P ;
                endif;
        endfor;
        return P;
end Selection_Projection;
```

2.9.

```
Algorithm  Join_Selection_Projection
Format: P = (R*S)(e,X)
Input:          - Quan hệ R(U)
                - Quan hệ S(V)
                - Biểu thức chọn e trên UV.
                - Tập con X của UV.
Output: - Quan hệ
P(X) = (R*S)(e,X) = {(u*v).X | u∈R,v∈S, u.M = v.M,
Sat(u*v,e), M = U ∩ V }
Method
```

```
        M := Attr(R) ∩ Attr(S);
        Create(P,X);
        for each tuple u in R do
            for each tuple v in S
                with u.M = v.M do
                    if Sat(u*v,e) then
                        construct t=(u*v).X;
                        if t not_in P then
                        add t to P ;
                        endif;
                    endif;
            endfor;
        endfor;
        return P;
    end Join_Selection_Projection;
```

2.10.

```
    Algorithm Card
    Format: c = Card(R)
    Input:  - Quan hệ R(U)
    Output: - số bộ của quan hệ R.
    Method
        c:=0;
        for each tuple u in R do
```

```
                c:=c+1;
            endfor;
            return c;
        end Card;
```

2.11.

```
        Algorithm Sum
        Format: s = Sum(R,A)
        Input:      - Quan hệ R(U)
                    - Thuộc tính kiểu số A trong U
        Output: - Tổng s các trị trên cột A trong
quan hệ R
        Method
            s:=0;
              for each tuple u in R do
                    s:=s+u.A;
              endfor;
              return s;
        end Sum;
```

2.12.

```
        Algorithm Avg
        Format: s = Avg(R,A)
        Input:      - Quan hệ R(U)
                    - Thuộc tính kiểu số A trong U
```

Output: - Trị trung bình của cột A trong quan hệ R

Method

```
s:=0;c:=0;
  for each tuple u in R do
        s:=s+u.A;
        c:=c+1;
  endfor;
  if c > 0 then return s/c
  else return null;
  endif;
end Avg;
```

2.13.

Algorithm Max

Format: smax = Max(R,A)

Input: - Quan hệ R(U)

- Thuộc tính kiểu sánh được A trong U

Output: - Trị lớn nhất của cột A trong quan hệ R

Method

```
smax:= -∞;
  for each tuple u in R do
```

```
                if smax < u.A then
                        smax:=u.A;
                endif;
        endfor;
        return smax;
end Max;
```

2.14.

```
Algorithm Min
Format: smin = Min(R,A)
Input:      - Quan hệ R(U)
            - Thuộc tính kiểu sánh được A
trong U
Output: - Trị nhỏ nhất của cột A trong quan
hệ R
Method
    smin:= +∞;
        for each tuple u in R do
                if smin > u.A then
                        smin:=u.A;
                endif;
        endfor;
        return smin;
end Min;
```

2.15.

Algorithm Sume

Format: s = Sume(R,A,e)

Input: - *Quan hệ R(U)*

- *Thuộc tính kiểu số A trong U*

- *Biểu thức e trên U*

Output: - Tổng s các trị trên cột A của các bộ thỏa điều kiện e trong quan hệ R

Method

```
s:=0;
  for each tuple u in R
        with sat(u,e) do
           s:= s+u.A;
  endfor;
  return s;
end Sume;
```

Bài giải Chương 3

NGÔN NGỮ HỎI SQL

CSDL *Thực tập*

SV(SV#, HT, NS, QUE,HL)

DT(DT#,TDT,CN,KP)

SD(SV#,DT#,NTT,KM, KQ)

3.1. Danh sách các sinh viên trẻ (dưới 18 tuổi) và học khá/giỏi (HL>8.5):

```
SELECT HT
FROM SV
WHERE 2003-NS < 18 AND HL > 8.5;
```

3.2. Thông tin về các đề tài được cấp kinh phí trên 10 triệu đồng:

```
SELECT *
FROM DT
WHERE KP > 10;
```

3.3. Danh sách các sinh viên trẻ (dưới 18 tuổi), học và thực tập đều đạt loại khá/giỏi (HL > 8.5 và KQ > 8.5):

```
SELECT HT
FROM SV
```

```
WHERE 2003-NS < 18 AND HL > 8.5
AND SV# IN
    (SELECT SV#
     FROM SD
     WHERE KQ > 8.5);
```

3.4. Danh sách các chủ nhiệm đề tài có các sinh viên quê ở Hà Nội tham gia:

```
SELECT CN
FROM DT
WHERE DT# IN
    (SELECT DT#
     FROM SD
     WHERE SV# IN
         (SELECT SV#
          FROM SV
          WHERE QUE = 'Ha Noi'));
```

3.5. Danh sách các sinh viên học giỏi hơn các sinh viên Hà Nội:

```
SELECT HT
FROM SV
WHERE HL > ALL
    (SELECT HL
```

```
    FROM SV
    WHERE QUE = 'Ha Noi');
```

3.6. Điểm trung bình của các sinh viên Hà Nội:

```
SELECT AVG(HL)
FROM SV
WHERE QUE = 'Ha Noi';
```

3.7. Tổng số đoạn đường thực tập theo đề tài 5:

```
SELECT SUM(KM)
FROM SD
WHERE DT# = 5;
```

3.8. Tổng số sinh viên đi thực tập:

```
SELECT COUNT(DISTINCT SV#)
FROM SD;
```

3.9. Số tỉnh có sinh viên đến thực tập theo đề tài 5:

```
SELECT COUNT(DISTINCT NTT)
FROM SD
WHERE DT# = 5;
```

3.10. Danh sách các tỉnh và số sinh viên quê ở tỉnh đó, nhóm theo QUE:

```
SELECT QUE, COUNT(*)
FROM SV
```

```
GROUP BY QUE;
```

3.11. Các đề tài có trên 10 sinh viên đăng ký tham gia:

```
SELECT TDT
FROM DT
WHERE 10 < (SELECT COUNT(*)
            FROM SD
            WHERE DT.DT# = SD.DT#);
```

***3.12.** Dùng SQL để biểu thị các phép toán của đại số quan hệ:

a) $R(e)$:

```
SELECT *
FROM R
WHERE e;
```

b) $R[X]$:

```
SELECT DISTINCT X
FROM R;
```

c) $R*S$: Giả sử $Attr(R) \cap Attr(S) = \{A_1, A_2, \ldots, A_k\}$

```
SELECT *
FROM R, S
WHERE R.A1 = S.A1
AND R.A2 = S.A2
AND...
```

```
AND R.Ak = S.Ak;
```

d) *R+S*:

```
R
UNION
S;
```

e) *R&S*:

```
R
INTERSECT
S;
```

f) *R-S*:

```
R
MINUS
S;
```

3.13. Cho thông tin về những sinh viên sinh trước năm 1973 và quê ở Hải Phòng:

```
SELECT *
FROM SV
WHERE NS < '1973 AND QUE = 'Hai Phong';
```

3.14. Cho danh sách các tỉnh có sinh viên đến thực tập:

```
SELECT DISTINCT NTT
FROM SD;
```

3.15. Cho biết các địa điểm thực tập xa trường (KM > 100) của đề tài số 7:

```
SELECT DISTINCT NTT
FROM SD
WHERE DT# = 7
    AND KM > 100;
```

3.16. Cho thông tin về việc thực tập tại Nha Trang của các sinh viên:

```
SELECT *
FROM SD
WHERE NTT = 'Nha Trang'
```

***3.17.** Cho danh sách sinh viên thực tập tại quê nhà:

```
SELECT HT
FROM SV, SD
WHERE SV.SV# = SD.SV#
     AND SV.QUE = SD.NTT;
```

3.18. Cho thông tin về các đề tài có sinh viên thực tập:

```
SELECT DISTINCT *
FROM DT
WHERE EXISTS
        (SELECT *
         FROM SD
         WHERE DT.DT# = SD.DT#);
```

3.19. Cho biết mã của các đề tài không có sinh viên nào tham gia:

```
SELECT DT#
FROM DT
MINUS
SELECT DISTINCT DT#
FROM SD;
```

3.20. Cho biết mã của những đề tài có kinh phí 1.5 triệu và những đề tài có kinh phí trên 2 triệu:

```
SELECT DT#
FROM DT
WHERE KP = 1.5
    OR KP > 2;
```

3.21. Cho biết mã của những sinh viên dưới 24 tuổi, thực tập khá (có điểm kết quả trên 6):

```
SELECT SV#
FROM SV
WHERE 2003 - NS < 24
   AND SV# IN
          (SELECT SV#
           FROM SD
           WHERE KQ > 6);
```

***3.22.** Cho danh sách các đề tài có sinh viên học giỏi nhất lớp tham gia:

```
SELECT TDT
FROM DT
WHERE DT# IN
    ( SELECT DISTINCT DT#
      FROM SD
      WHERE SV# IN
        ( SELECT SV#
          FROM SV
          WHERE HL = ( SELECT MAX(HL)
                       FROM SV )));
```

***3.23.** Cho danh sách các đề tài không có sinh viên học kém nhất lớp tham gia:

```
SELECT TDT
FROM DT
WHERE DT# IN
      (( SELECT DT#
         FROM DT )
      MINUS
      ( SELECT DISTINCT DT#
```

```
        FROM SD
        WHERE SV# IN
           ( SELECT SV#
             FROM SV
             WHERE HL = ( SELECT MIN(HL)
                          FROM SV ))));
```

Chú ý: biểu thức SQL sau đây

```
SELECT TDT
FROM DT
WHERE DT# IN
    ( SELECT  DISTINCT DT#
      FROM SD
      WHERE SV# IN
         ( SELECT SV#
           FROM SV
           WHERE HL > ( SELECT MIN(HL)
                        FROM SV )));
```

cho biết danh sách các đề tài có sinh viên tham gia, nhưng những sinh viên này không phải là những người học kém nhất lớp.

***3.24.** Cho danh sách những sinh viên thực tập theo đề tài có kinh phí lớn hơn một phần năm tổng kinh phí cấp cho các đề tài:

```
SELECT HT
FROM SV
```

```
WHERE SV# IN
    ( SELECT SV#
      FROM SD
      WHERE DT# IN
        ( SELECT DT#
          FROM DT
          WHERE 5*KP > ( SELECT SUM(KP)
                         FROM DT )));
```

*3.25. Cho danh sách các sinh viên có điểm học tập cao hơn điểm thực tập trung bình của đề tài mã số 4:

```
SELECT HT
FROM SV
WHERE HL > ( SELECT AVG(KQ)
             FROM SD
             WHERE DT# = 4);
```

*3.26. Cho quan hệ *R(U)*. Hãy dùng SQL để sinh ra quan hệ rỗng *S(U)*:

```
SELECT *
FROM R
WHERE False;
```

Bài giải Chương 4
PHỤ THUỘC HÀM

4.1.

f_1: $A \rightarrow A$: *thỏa,*

f_2: $A \rightarrow B$: *thỏa,*

f_3: $A \rightarrow C$: *không thỏa,*

f_4: $AC \rightarrow C$: *thỏa,*

f_5: $A \rightarrow D$: *thỏa,*

f_6: $D \rightarrow A$: *thỏa.*

4.2.

```
Algorithm Closure
Format: Y = X+
Input:      - LĐQH p =(U,F)
            - Tập thuộc tính X ⊆ U
Output: - Y = X+ = {A ∈ U | X → A ∈ F+}
Method
        Y:=X;
    repeat
            Z:=Y;
            for each FD L→R in F do
```

```
                    if L ⊆ Y then
                        Y:=Y ∪ R;
                    endif;
                endfor;
            until Y=Z;
            return Y;
    end Closure;
```

4.3.

F1. Nếu $Y \subseteq X$ thì $R(X \rightarrow Y)$ (*tính phản xạ*)

$\forall u,v \in R$: $u.X=v.X \Rightarrow u.Y=v.Y$, vì $Y \subseteq X$.

F2. Nếu $R(X \rightarrow Y)$ thì $R(XZ \rightarrow YZ)$ (*tính gia tăng*)

$\forall u,v \in R$: $u.XZ=v.XZ \Rightarrow u.X=v.X$ & $u.Z=v.Z \Rightarrow u.Y=v.Y$ & $u.Z=v.Z \Rightarrow u.YZ=v.YZ$

F3. Nếu $R(X \rightarrow Y)$ và $R(Y \rightarrow Z)$ thì $R(X \rightarrow Z)$ (*tính bắc cầu*)

$\forall u,v \in R$: $u.X=v.X \Rightarrow u.Y=v.Y \Rightarrow u.Z=v.Z$

4.4. Giả sử f: $X \rightarrow Y \notin F^+$. Ta chứng minh $X \rightarrow Y \notin F^+$ bằng cách chỉ ra một quan hệ Armstrong $R(U)$ thoả các PTH trong tập F (thậm chí trong F^+) nhưng không thoả PTH f.

Quan hệ Armstrong R được xây dựng như sau:

Giả sử $U = \{A_1, A_2, ..., A_n\}$ và a_i và b_i là hai phần tử khác nhau của $dom(A_i)$, $i = 1..n$. Quan hệ R chứa 2 bộ u và v như sau:

$u = (a_1, a_2, ..., a_n)$.

$v.A_i=a_i$, nếu $A_i \in X^+$, nếu không ta đặt $v.A_i=b_i$, $i = 1..n$.

Trước hết ta chứng minh R không thoả PTH $X \to Y$. Theo cách xây dựng R, ta có hai bộ u và v giống nhau trên miền lớn duy nhất là X^+, $u.X^+ = v.X^+$ và do $X^+ \supseteq X$ nên $u.X = v.X$. Giả sử $u.Y = v.Y$. Thế thì $Y \subseteq X^+$. Theo định nghĩa bao đóng ta suy ra $X \to Y \in F^+$, mâu thuẫn với giả thiết. Vậy R không thoả PTH $X \to Y$.

Ta chứng minh R thoả mọi PTH trong F^+. Giả sử $W \to Z \in F^+$ và $u.W = v.W$. Từ đây rút ra, do đặc điểm của R, $W \subseteq X^+$. Theo định nghĩa bao đóng, $X \to W \in F^+$, theo tính chất bắc cầu cho các PTH $X \to W$ và $W \to Z$ ta suy ra $X \to Z \in F^+$. Lại theo định nghĩa bao đóng ta có $Z \subseteq X^+$ và do đó, theo đặc điểm của R ta có $u.Z = v.Z$. Vậy R thoả $W \to Z$ *đpcm*.

Chú ý: *Chứng minh được dựa trên giả thiết là miền trị của các thuộc tính trong quan hệ chứa ít nhất 2 trị phân biệt. Giả thiết này là khá tự nhiên, vì nếu trong bảng có một cột chỉ chứa một trị duy nhất thì ta có thể xoá cột đó.*

4.5. a. Nếu $F \subseteq G$ thì $SAT(F) \supseteq SAT(G)$: $R(G) \Rightarrow R(F)$.

b. $SAT(FG) = SAT(F) \cap SAT(G)$: $R(FG) \Leftrightarrow R(F) \wedge R(G)$.

4.6. ***Chú ý:*** *Những chỗ có dấu ? là gợi ý bạn đọc giải thích vì sao.*

Chứng minh mệnh đề b trước sau đó suy ra mệnh đề a.

b. $R \subseteq S \Rightarrow FD(R) \supseteq FD(S)$:

Giả sử $X \to Y \in FD(S)$ và u và v là hai bộ trong R thỏa $u.X = v.X$. Ta có $u, v \in S$ (?). Do đó $u.Y = v.Y$ (?), từ đó suy ra $X \to Y \in FD(R)$.

a. $FD(R+S) \subseteq FD(R) \cap FD(S)$

Vì $R+S \supseteq R$ và $R+S \supseteq S$ nên, theo câu *b* ta có

$FD(R+S) \subseteq FD(R)$ và $FD(R+S) \subseteq FD(S)$. Từ đó suy ra a.

Thí dụ chứng tỏ $FD(R+S) \subset FD(R) \cap FD(S)$.

$U=AB$; R chứa một bộ duy nhất $u = (1,x)$; S chứa một bộ duy nhất $v = (1,y)$, $x \neq y$. R và S thỏa mọi PTH trên $U(?)$. Quan hệ $P=R+S$ chứa 2 bộ u và v. P không thỏa PTH $A \rightarrow B(?)$.

4.7.

Nhận xét : Các toán tử *SAT* và *FD* có tính nghịch biến(?) và

$$SAT(F) = \bigcap_{f \in F} SAT(f) \quad \text{và}$$

$$FD(\Re) = \bigcap_{R \in \Re} FD(R)$$

4.8.

Với mọi tập con X, Y, Z, V của U và với mọi thuộc tính A trong U:

F4. Tính tựa bắc cầu: Nếu $X \rightarrow Y$, $YZ \rightarrow V$ thì $XZ \rightarrow V$

$X \rightarrow Y$ (gt)

$YZ \rightarrow V$ (gt)

$XZ \rightarrow YZ$ (F2)

$XZ \rightarrow V$ (F3)

F5. Tính phản xạ chặt: $X \rightarrow X$ (F1)

F6. Mở rộng vế trái và thu hẹp vế phải: Nếu $X \rightarrow Y$ thì $XZ \rightarrow Y \setminus V$

$X \rightarrow Y$ (gt)

$Y \rightarrow Y \setminus V$ (F1)

$X \rightarrow Y \setminus V$ (F3)

$XZ \rightarrow X$ (F1)

$XZ \rightarrow Y \setminus V$ (F3)

F7. Cộng tính đầy đủ: Nếu $X \rightarrow Y$ và $Z \rightarrow V$ thì $XZ \rightarrow YV$

$X \rightarrow Y$ (gt)

$Z \rightarrow V$ (gt)

$XZ \rightarrow YZ$ (F2)

$YZ \rightarrow YV$ (F2)

$XZ \rightarrow YV$ (F3)

F8. Mở rộng vế trái: Nếu $X \rightarrow Y$ thì $XZ \rightarrow Y$

$XZ \rightarrow X$ (F1)

$X \rightarrow Y$ (gt)

$XZ \rightarrow Y$ (F3)

F9. Cộng tính ở vế phải: Nếu $X \rightarrow Y$ và $X \rightarrow Z$ thì $X \rightarrow YZ$

$X \rightarrow Y$ (gt)

$X \rightarrow XY$ (F2)

$X \rightarrow Z$ (gt)

$XY \rightarrow YZ$ (F2)

$X \rightarrow YZ$ (F3)

F10. Bộ phận ở vế phải: Nếu $X \rightarrow YZ$ thì $X \rightarrow Y$

$X \rightarrow YZ$ (gt)

$YZ \rightarrow Y$ (F1)

$X \rightarrow Y$ (F3)

F11.Tính tích luỹ: Nếu $X \rightarrow YZ$, $Z \rightarrow AV$ thì $X \rightarrow YZA$

$Z \rightarrow AV$ (gt)

$YZZ \rightarrow YZAV$ (F2)

$YZ \rightarrow YZAV$

$YZAV \rightarrow YZA$ (F1)

$YZ \rightarrow YZA$ (F3)

$X \rightarrow YZ$ (gt)

$X \rightarrow YZA$ (F3)

4.9.

$\forall X, Y \in Poset(U)$

(C4) $f(XY) \supseteq f(X)f(Y)$:

$XY \supseteq X$ (ltth- theo lý thuyết tập hợp)

$f(XY) \supseteq f(X)$ (C2)

$XY \supseteq Y$ (ltth)

$f(XY) \supseteq f(Y)$ (C2)

$f(XY) \supseteq f(X)f(Y)$ (Cộng tính của ltth)

(C5) $f(X \cap Y) \subseteq f(X) \cap f(Y)$:

$X \cap Y \subseteq X$ (ltth)

$f(X \cap Y) \subseteq f(X)$ (C2)

$X \cap Y \subseteq Y$ (ltth)

$f(X \cap Y) \subseteq f(Y)$ (C2)

$f(X \cap Y) \subseteq f(X) \cap f(Y)$ (ltth)

(C6) $f(f(X)Y) = f(XY)$:

$X \subseteq f(X)$ (C1)

$XY \subseteq f(X)Y$ (ltth)

$f(XY) \subseteq f(f(X)Y)$ (C2)

$X \subseteq XY$ (ltth)

$f(X) \subseteq f(XY)$ (C2)

$Y \subseteq XY$ (ltth)

$XY \subseteq f(XY)$ (C1)

$Y \subseteq f(XY)$ (Tính chất bắc cầu của ltth)

$f(X)Y \subseteq f(XY)$ (Cộng tính của ltth)

$f(f(X)Y) \subseteq f(f(XY))=f(XY)$ (C2,C3)

$f(Xf(Y)) = f(XY)$: tương tự

Chứng minh phép toán lấy bao đóng của tập thuộc tính là một ánh xạ đóng thỏa các tính chất sau:

1. Tính phản xạ $X^+ \supseteq X$: (đnbđ - Theo định nghĩa bao đóng)
2. Tính đơn điệu nếu $X \subseteq Y$ thì $X^+ \subseteq Y^+$ (ttbđ - thuật toán tìm bao đóng)
3. Tính luỹ đẳng $X^{++} = X^+$: (ttbđ)
4. $(XY)^+ \supseteq X^+Y^+$: (C4)
5. $(X^+Y)^+ = (XY^+)^+ = (XY)^+$: (C6)
6. $X \rightarrow Y$ khi và chỉ khi $Y \subseteq X^+$: (đnbđ)

7. $X \to Y$ khi và chỉ khi $Y^+ \subseteq X^+$: (đnbđ, C2,C3)

8. $X \to X^+$ và $X^+ \to X$: (đnbđ,C1)

9. $X^+ = Y^+$ khi và chỉ khi $X \to Y$ và $Y \to X$: (đnbđ)

4.10.

$A^o = \{F1, F2, F3\} \Leftrightarrow B^o = \{F5, F10, F11\}$

$[A^o \Rightarrow B^o]$: *xem bài 4.8*

$[B^o \Rightarrow A^o]$

$[B^o \Rightarrow F1]$:

$X \supseteq Y$ (gt)

$X = MY$ (gt)

$MY \to MY$ (F5)

$X \to MY$

$X \to Y$ (F10), đpcm.

Trước hết chứng minh $[B^o \Rightarrow F11']$ với

F11' (*Tính tích luỹ mở rộng*): Nếu $X \to YZ$ và $Z \to MV$ thì $X \to YZM$ trong đó M là một tập con của U.

Giả sử $M = A_1A_2...A_k$. Ta ký hiệu $V_i = (M \backslash A_i)V$, $i = 1..k$. Ta có, với mọi $i = 1..k$: $MV = A_iV_i$.

$X \to YZ$ (gt)

$Z \to A_1V_1$ (gt)

$X \to YZA_1$ (F11)

$X \to (YA_1)Z$

$Z \to A_2V_2$ (gt)

$X \to YA_1ZA_2$ (F11)

$X \to (YA_1A_2)Z$

...

$X \to (YA_1A_2 \ldots A_k)Z$ (F11), hay

$X \to YZM$, đpcm.

[$B^o \Rightarrow$ F3]:

$X \to \varnothing Y$ (gt)

$Y \to \varnothing Z$(gt)

$X \to \varnothing YZ$ (F11')

$X \to Z$ (F10), đpcm.

[$B^o \Rightarrow$ F2]:

$XZ \to XZ$ (F5)

$X \to Y$ (gt)

$XZ \to XYZ$ (F11')

$XYZ \to YZ$ (F1 đã chứng minh)

$XZ \to YZ$ (F3 đã chứng minh), đpcm.

4.11. $A^o = \{F1, F2, F3\} \Leftrightarrow S^o = \{F1, F4\}$

[$A^o \Rightarrow S^o$]: *xem bài 4.8*

[$S^o \Rightarrow A^o$]:

[$S^o \Rightarrow$ F2]:

$X \rightarrow Y$ (gt)

$YZ \rightarrow YZ$ (F1)

$XZ \rightarrow YZ$ (F4), đpcm

[$S^o \Rightarrow$ F3]:

$X \rightarrow Y$ (gt)

$Y\varnothing \rightarrow Z$ (gt)

$X\varnothing \rightarrow Z$ (F4)

$X \rightarrow Z$, đpcm.

4.12. A^o = {F1, F2, F3} $\Leftrightarrow$ D^o = {F3, F5, F6, F7}

[$A^o \Rightarrow D^o$]: *xem bài 4.8*

[$D^o \Rightarrow A^o$]:

[$D^o \Rightarrow$ F1]:

$X \supseteq Y$ (gt)

$X = MY$ (gt)

$Y \rightarrow Y$ (F5)

$MY \rightarrow Y - \varnothing$ (F6)

$X \rightarrow Y$, đpcm.

[$D^o \Rightarrow$ F2]:

$X \rightarrow Y$ (gt)

$Z \rightarrow Z$ (F5)

$XZ \rightarrow YZ$ (F7), đpcm.

4.13. $A^o = \{F1, F2, F3\} \Leftrightarrow M^o = \{F4, F5, F8\}$

[$A^o \Rightarrow M^o$]: *xem bài 4.8*

[$M^o \Rightarrow A^o$]:

[$M^o \Rightarrow F1$]:

$X \supseteq Y$ (gt)

$X = MY$ (gt)

$Y \rightarrow Y$ (F5)

$MY \rightarrow Y$ (F8)

$X \rightarrow Y$, đpcm.

[$M^o \Rightarrow F2$]:

$X \rightarrow Y$ (gt)

$YZ \rightarrow YZ$ (F5)

$XZ \rightarrow YZ$ (F4), đpcm.

[$M^o \Rightarrow F3$]:

$X \rightarrow Y$ (gt)

$Y \rightarrow Z$ (gt)

$Y\varnothing \rightarrow Z$ (gt)

$X\varnothing \rightarrow Z$ (F4)

$X \rightarrow Z$, đpcm.

4.14. $p = (U,F)$, $U = ABCDE$, $F = \{A \rightarrow C, BC \rightarrow D, D \rightarrow E, E \rightarrow A\}$.

a) $(AB)^+ = ABCDE = U$.

b) $(BD)^+ - D^+ = ABCDE-ACDE= B$.

4.15. $p = (U,F)$, $U = ABCDEG$, $F = \{ B \to C, Y \to D, D \to G, AG \to E \}$.

a) $AB \to G \in F^+$ vì $(AB)^+ = ABCDEG \supseteq G$

b) $BD \to AD \notin F^+$ vì $AD \not\subset (BD)^+ = BCDG$.

4.16 $F \equiv G \Leftrightarrow (\forall X \subseteq U): (X_F^+ = X_G^+)$?

$X \to Y \in F^+ \Leftrightarrow Y \subseteq X_F^+ = X_G^+ \Leftrightarrow X \to Y \in G^+$.

4.17 $F \equiv F^+$?

$F \models F^+$ (theo định nghĩa)

$F^+ \models F$ (vì $F^+ \supseteq F$)

4.18

a) $F=\{ X\to Y, Z \to V \} \models G=\{ XZ \to YV \}$, vì $(XZ)^+_F=XYZV \supseteq YV$.

b) $F=\{ X\to Y \} \equiv G=\{ X\to Y-X \}$, vì $X^+_F = X_G^+ = XY$.

c) $F=\{ X\to Y \} \models G=\{ XZ\to Y \}$, vì $(XZ)^+_F = XYZ \supseteq Y$.

d) $F=\{ X\to Y , Y \to Z \} \models G= \{ X\to Z \}$, vì $X^+_F = XYZ \supseteq Z$.

e) $F=\{ X\to Y, YZ\to V \} \models G=\{ XZ\to V \}$, vì $(XZ)^+_F = XYZV \supseteq V$.

f) $F=\{ X \to Y \} \models G=\{ XZ\to Y-V \}$, vì $(XZ)^+_F = XYZ \supseteq Y \supseteq Y-V$.

g) $F=\{X \to Y, X \to Z\} \equiv G=\{X\to YZ\}$, vì $X^+_F = X^+_G = XYZ \supseteq Y,Z,YZ$.

h) $F=\{ X \to YZ \} \models G=\{ X\to Y \}$, vì $X^+_F = XYZ \supseteq Y$.

i) $F=\{ X\to YZ, Z\to AV \} \models G=\{ X\to YZA \}$, vì $X^+_F = XYZAV \supseteq YZA$.

4.19 Thuật toán tìm phủ thu gọn tự nhiên của tập PTH F.

```
Algorithm Natural_Reduced
```

```
Format: Natural_Reduced(F)
Input:  - Tập PTH F
Output: - Một phủ thu gọn tự nhiên G của F
```

- $G \equiv F$
- $\forall L \to R \in G$: $L \cap G = \varnothing$
- $\forall L_i \to R_i, \forall L_j \to R_j \in G$: $i \neq j \Rightarrow L_i \neq L_j$

```
Method
      G:=∅;
    for each FD L → R in F do
            Z:=R-L;
          if Z ≠ ∅ then
            if there is an FD L→Y in G then
                replace L→Y in G by L→YZ
              else add L→Z to G;
             endif;
          endif;
    endfor;
    return G;
  end Natural_Reduced;
```

4.20 *Thuật toán tìm phủ không dư của tập PTH F.*

```
Algorithm Nonredundant
```

```
Format: Nonredundant(F)
Input:        - Tập PTH F
Output: - Một phủ không dư G của F
   •  G ≡ F
   •  ∀ g ∈G: G-{g} !≡ G
Method
      G:=F;
    for each FD g:L→R in F do
        if R ⊆ L+G-{g}  then
          G:= G -{g};
        endif;
   endfor;
   return G;
end Nonredundant;
```

4.21. *Xây dựng thuật toán tìm phủ thu gọn trái của tập PTH F.*

Để ý rằng ta luôn có

$\forall\ g{:}L\to R\ \in G, \forall A\in L{:}\ G-\{g\}\cup\{L-\{A\}\to R\} \models L\to R,$

vì $L-\{A\}\subseteq L$, *do đó ta chỉ cần kiểm tra*

$G \models (L-\{A\})\to R.$

```
Algorithm Left_Reduced
      Format: Left_Reduced(F)
```

```
Input:        - Tập PTH F
Output: - Một phủ thu gọn trái G của F
   • G ≡ F
   • ∀ g:L→R ∈G, ∀A∈L: G-{g}∪{L-{A}→R} !≡G
Method
      G:=F;
    for each FD g:L→R in F do
        X:=L;
        for each attribute A in X do
                  if R ⊆(L-{A})⁺G then
              delete A from L in G;
          endif;
        endfor;
    endfor;
    return G;
  end Left_Reduced;
```

4.22. *Xây dựng thuật toán tìm phủ thu gọn phải của tập PTH F.*

Để ý rằng ta luôn có

$\forall g: L \to R \in G, \forall A \in R: G \models L \to R-\{A\}$,

vì $R-\{A\} \subseteq R$, do đó ta chỉ cần kiểm tra

$G-\{L \to R\} \cup \{L \to R-\{A\}\} \models L \to R$.

```
Algorithm Right_Reduced
    Format: Right_Reduced(F)
    Input:      - Tập PTH F
    Output: - Một phủ thu gọn phải G của F
      • G ≡ F
      • ∀ g:L→R ∈G, ∀A∈R: G-{g}∪{L→R-{A}} !≡G
    Method
        G:=F;
      for each FD g:L → R in F do
          X:=R;
          for each attribute A in X do
                  if A in L+ G-{L->R}∪{L->R-{A}}  then
              delete A from R in G;
            endif;
          endfor;
      endfor;
      return G;
    end Right_Reduced;
```

4.23. Xây dựng thuật toán tìm phủ thu gọn của tập PTH F.

```
Algorithm Reduced
    Format: Reduced(F)
```

```
Input:        - Tập PTH F
Output: - Một phủ thu gọn G của F
Method
        G:=Right_Reduced(Left_Reduced(F));
    return G;
end Reduced;
```

4.24. Xây dựng thí dụ chứng tỏ với tập PTH F sau khi thực hiện

```
G:=Left_Reduced(Right_Reduced(F));
```

G lại có thể trở thành phủ chưa thu gọn phải.

Xét tập PTH

$F = \{DA \to BC, B \to CD, A \to D\}$

$H := Right\_Reduced(F) = \{DA \to B, B \to CD, A \to D\}$

$G := Left\_Reduced(H) = \{A \to B, B \to CD, A \to D\}$

G không phải là tập PTH thu gọn phải vì ta còn có thể bỏ thuộc tính D ở PTH thứ ba trong G,

$Right\_Reduced(G) = \{A \to B, B \to CD, A \to \varnothing\} \equiv \{A \to B, B \to CD\}$

Chú ý rằng các PTH có vế phải là tập rỗng thì không chứa thông tin do đó có thể loại bỏ chúng khỏi tập PTH. Từ đó suy ra rằng nếu tập PTH có các vế phải là một thuộc tính thì việc tìm thu gọn phải tương đương với việc tìm phủ không dư.

4.25. Thuật toán tìm phủ tối tiểu của tập PTH F.

```
Algorithm MinCover
```

```
Format: MinCover(F)
Input:      - Tập PTH F
Output: - Một phủ tối tiểu G của F
Method
        // Tách mỗi PTH L→R trong F thành
        // các PTH L→A, A ∈ R
         G := Ø;
        for each FD L → R in F do
          for each attribute A in R do
         if L→A not_in G then
            add L→A to G;
         endif;
      endfor;
    endfor;
   G:=Reduced(G);
   return G;
  end MinCover;
```

4.26. Chứng minh rằng với mọi tập PTH F trên *U* luôn tồn tại một phủ G của *F* sao cho mọi PTH trong G đều là phụ thuộc đầy đủ:

Đó chính là phủ thu gọn trái của F(?).

4.27. Xây dựng thuật toán tìm một phủ đầy đủ của tập PTH *F*: *Xem bài 4.21.*

4.28. Cho quan hệ R trên U và các tập con thuộc tính X, Y của U. Chứng minh:

a) $R(X\ (s) \rightarrow Y) \Rightarrow R(X \rightarrow Y)$

$\forall\ u,v \in R,\ u.X = v.X$ (gt)

$u.Y = v.Y$ (vì $R(X\ (s) \rightarrow Y)$)

$R(X \rightarrow Y)$, đpcm

b) $R(X \rightarrow Y) \Rightarrow R(X\ (w) \rightarrow Y)$

$\forall\ u,v \in R$: $u.X = v.X$ (gt)

$u.Y = v.Y$ (vì $R(X \rightarrow Y)$)

$R(X\ (w) \rightarrow Y)$, đpcm

c) $R(X\ (d) \rightarrow Y) \Rightarrow R(X\ (w) \rightarrow Y)$

$\forall\ u,v \in R,\ u.X = v.X$ (gt)

$\exists\ B \in Y$: $u.B = v.B$ (vì $R(X\ (d) \rightarrow Y)$)

$R(X\ (w) \rightarrow Y)$, đpcm

d) $R(X\ (s) \rightarrow Y) \Rightarrow R(X\ (d) \rightarrow Y)$

$\forall\ u,v \in R,\ \exists\ A \in X$: $u.A = v.A$ (gt)

$u.Y = v.Y$ (vì $R(X\ (s) \rightarrow Y)$)

$R(X\ (d) \rightarrow Y)$, đpcm

4.29. Với mọi tập con thuộc tính X, Y, Z và thuộc tính A trên tập thuộc tính U. Điền các ký hiệu f, s, w hoặc d thay cho dấu ? để khẳng định các tính chất cao nhất của các phụ thuộc mạnh, yếu hoặc đối ngẫu sau đây:

1. Tính phản xạ: Nếu $Y \subseteq X$ thì $X\ (f) \rightarrow Y$

2. Tính gia tăng: Nếu $X\ (f) \rightarrow Y$ thì $XZ\ (f) \rightarrow YZ$

3. Tính bắc cầu: Nếu $X\ (s) \rightarrow Y$ và $Y\ (s) \rightarrow Z$ thì $X\ (s) \rightarrow Z$

4. Tính tựa bắc cầu: Nếu $X\ (s) \rightarrow Y$, $YZ\ (s) \rightarrow V$ thì $XZ\ (s) \rightarrow V$

5. *Tính phản xạ chặt:* $X\ (d) \rightarrow X$ và $X\ (f) \rightarrow X$

6. *Mở rộng vế trái và thu hẹp vế phải:*

   Nếu $X\ (f) \rightarrow Y$ *thì* $XZ\ (f) \rightarrow Y \setminus V$

7. Cộng tính đầy đủ: Nếu $X\ (f) \rightarrow Y$ và $Z\ (f) \rightarrow V$ thì $XZ\ (f) \rightarrow YV$

8. Mở rộng vế trái: Nếu $X\ (f) \rightarrow Y$ thì $XZ\ (f) \rightarrow Y$

9. Cộng tính ở vế phải: Nếu $X\ (s) \rightarrow Y$ và $X\ (s) \rightarrow Z$ thì $X\ (s) \rightarrow YZ$

10. Bộ phận ở vế phải: Nếu $X\ (s) \rightarrow YZ$ thì $X\ (s) \rightarrow Y$.

11. Tính tích luỹ: Nếu $X\ (s) \rightarrow YZ$, $Z\ (s) \rightarrow AV$ thì $X\ (s) \rightarrow YZA$

4.30. Xây dựng thuật toán tìm một khóa của LĐQH.

Tư tưởng: *Xuất phát từ một siêu khóa K tùy ý của LĐQH, duyệt lần lượt các thuộc tính A của K, nếu bất biến* $(K-\{A\})^+ = U$ *được bảo toàn thì loại A khỏi K. Có thể thay kiểm tra* $(K-\{A\})^+ = U$ *bằng kiểm tra* $A \in (K-\{A\})^+$ *(?).*

```
Algorithm Key

    Format: Key(U,F)

    Input:      - Tập thuộc tính U

                - Tập PTH F

    Output: - Khóa K ⊆ U thỏa
```

- $K^+ = U$
- $\forall A \in K$: $(K-\{A\})^+ \neq U$

```
Method
        K:=U;
      for each attribute A in U do
            if A ∈(K-{A})⁺ then
                K := K-{A}
            endif;
      endfor;
      return K;
end Key;
```

4.31. Cho LĐQH p. Biết p có một khóa K. Hãy xây dựng thuật toán tìm một khóa thứ hai M của p. Nếu p không có khóa thứ hai thuật toán cho kết quả là một tập rỗng.

Tư tưởng: *Xuất phát từ một siêu khóa M tùy ý, trước hết duyệt các thuộc tính A của K, nếu bất biến $(M-\{A\})^+ = U$ được bảo toàn thì loại A khỏi M. Sau đó duyệt tương tự với các thuộc tính trong U-K.*

```
Algorithm Key2
      Format: Key2(U,F)
      Input:      - Tập thuộc tính U
                  - Tập PTH F
                  - Khóa K ⊆ U
```

Output: - Khóa thứ hai, nếu có, M $\subseteq$ U thỏa

- $M^+ = U$
- $\forall A \in M$: $(M-\{A\})^+ \neq U$

Nếu không có khóa thứ hai: $\varnothing$.

```
Method
      M:=U;
    for each attribute A in K do
          if A ∈(M -{A})⁺ then
             M := M -{A}
          endif;
   endfor;
   for each attribute A in U-K do
          if A ∈(M -{A})⁺ then
             M := M -{A}
          endif;
   endfor;
   if M = K then  return ∅
     else return M;
   endif
end Key2;
```

4.32. Xây dựng một LĐQH có 5 thuộc tính ABCDE, mỗi thuộc tính là một khóa:

p = (U,F); U = ABCDE; F = {A→B, B→C, C→D, D→E, E→A}.

Ta có $\forall x \in U$: $x^+ = U$ (?), do đó *x* là một khóa của *p*.

Chú ý: *Tập PTH F có dạng như trên được gọi là tập phụ thuộc hàm dạng vòng.*

4.33. LĐQH có 5 thuộc tính có thể có tối đa bao nhiêu khóa. Cho thí dụ.

10 khóa, thí dụ, p = (U,F); U = ABCDE; F = {AB→CDE, AC→BDE, AD→BCE, AE→BCD, BC→ADE, BD→ACE, BE→ACD, CD→ABE, CE→ABD, DE→ABC}. Dễ thấy mỗi vế trái tạo ra một khóa (?)

4.34. Xây dựng một LĐQH có 5 thuộc tính ABCDE và chỉ có một khóa duy nhất. p = (U,F); U = ABCDE; F = {A→BCDE}.

4.35. (*Nguyễn Xuân Huy*) Cho K là một khóa của LĐQH p = (U,F). Chứng minh rằng với mọi tập con X của K ta có: $X^+ \cap K = X$.

Vì $X \subseteq X^+$ và $X \subseteq K$ nên $X \subseteq X^+ \cap K$. Ta cần chứng minh $X^+ \cap K \subseteq X$. Giả sử $A \in X^+ \cap K$ và $A \notin X$. Ta xét tập M = K-{A}. Dễ thấy $X \subseteq M$ (?). Ta có, theo tính chất đồng biến của bao đóng: $A \in X^+ \subseteq M^+$

Từ đây suy ra $M^+ \supseteq K$ nên $M^+ = U$(?), tức là M là bộ phận thực sự của khóa K lại đồng thời là siêu khóa, trái với định nghĩa khóa. Vậy $A \in X$. đpcm.

Chú ý: *Tính chất trên là một đặc trưng của các thuộc tính nguyên thủy (thuộc tính khóa).*

4.36. (*Lê Văn Bào, Nguyễn Xuân Huy, Hồ Thuần*) Cho LĐQH p = (U,F). Gọi M là giao của các khóa của p. Chứng minh rằng

$$M = U - \bigcup_{L \to R \in F} (R - L)$$

Chứng minh

Trước hết để ý rằng các PTH L→R và L→(R-L) là tương đương do đó ta có thể giả thiết rằng mọi PTH trong F đều có dạng L→R, L∩ R=∅ (xem thêm bài 4.19). Do giả thiết này ta có R-L=R. Dễ nhận thấy M là tập các thuộc tính không có mặt trong vế phải của mọi PTH trong F (?) do đó chúng phải có mặt trong mọi khóa (?). Giả sử A là một thuộc tính có trong vế phải của PTH L→AR' của F. Ta chứng minh A sẽ không xuất hiện trong một khóa K nào đấy của p. Thật vậy, xét tập X = U-A. Dễ thấy X ⊇ L (?) và X là siêu khóa (?). Từ siêu khóa X không chứa A (?) ta lấy ra được một khóa K. K không chứa A (?).

4.37. (*Lê Văn Bào, Hồ Thuần*)Cho LĐQH p = (U,F). Gọi M là giao của các khóa của p. Chứng minh rằng p có một khóa duy nhất khi và chỉ khi $M^+ = U$.

Chứng minh:

Nếu $M^+ = U$ thì M là siêu khóa và M không thể chứa thực sự một khóa (?), tức là M là một khóa. Vì M là giao của các khóa đồng thời lại là khóa nên p không thể còn khóa nào khác ngoài M. Ngược lại, nếu p chỉ có một khóa duy nhất M thì giao của các khóa đương nhiên là M, và do đó, theo tính chất của khóa $M^+ = U$.

4.38. Cho tập thuộc tính U với n phần tử. Chứng minh rằng có thể xây dựng tập PTH *F* sao cho LĐQH *p* = (*U*,*F*) có

$$C_n^{\lfloor \frac{n}{2} \rfloor} \text{ khóa,}$$

trong đó toán tử $\lfloor x \rfloor$ cho ta cận nguyên dưới của số nguyên *x*,

C_n^m là tổ hợp chập m của n phần tử.

Gợi ý: Giả sử $U = \{A_1, A_2, ..., A_n\}$, $n > 0$. *Xây dựng tập PTH* $F = \{L_i \rightarrow U - L_i \mid i = 1...m\}$, *trong đó mỗi* L_i *là một tổ hợp chập* $\lfloor n / 2 \rfloor$ *của n. Chứng ming rằng mỗi vế trái* L_i *là một khóa.*

4.39. Tìm tập thuộc tính nguyên thuỷ của LĐQH sau:

p = (*U*,*F*), *U* = *ABCDE*,

$F = \{ AB \rightarrow C, AD \rightarrow B, B \rightarrow D \}$.

Lược đồ p có giao các khóa là: AE, mà $(AE)^+ = AE \neq U$ *nên p có trên 1 khóa, cụ thể là 2 khóa: ABE và ADE. Vậy tập thuộc tính nguyên thủy là* $U_k = ABE \cup ADE = ABDE$.

4.40. Khẳng định hoặc bác bỏ tính đúng của các mệnh đề sau:

a) $K \subseteq U$ là một khoá khi và chỉ khi $K \mathrel{+\!\!\rightarrow} U$ (phụ thuộc đầy đủ): *đúng.*

b) Hai khoá khác nhau của một LĐQH không giao nhau: *sai, xem bài 4.39.*

c) Hai khoá khác nhau của một LĐQH không bao nhau: *đúng.*

d) Mọi LĐQH đều có ít nhất một khoá: *đúng.*

e) Tồn tại một LĐQH không có khoá nào: *sai.*

f) Số khoá của một LĐQH không thể lớn hơn số thuộc tính: *sai, xem bài 4.33.*

g) U không thể là khoá của LĐQH (U,F): *sai; khi F = ∅ hoặc F chỉ chứa các PTH tầm thường thì U là khóa.*

h) Mọi LĐQH không thể có hai khoá đơn tức là khoá chỉ gồm một thuộc tính: *sai, xem bài 4.32.*

Bài giải chương 5
CHUẨN HÓA

5.1. Chứng minh rằng nếu $X \to Y$ và $X \cap Y = \varnothing$ thì phép tách (XY, U-Y) là không tổn thất.

Chứng minh

Đặt $M=XY$, $Z=U-Y$, ta có $MZ=U$ và $M \cap Z = X$. Ta cần chứng minh $\forall R \in SAT(X \to Y)$: $R[M]*R[Z]=R$. Theo bài 1.33 ta có $R[M]*R[Z] \supseteq R$, ta còn phải chứng minh $R[M]*R[Z] \subseteq R$. Giả sử $u \in R[M]$, $v \in R[Z]$ và u và v thoả điều kiện kết nối tự nhiên, tức là $u.X = v.X$, ta chứng minh $u*v \in R$. Vì $u \in R[M]$ nên $\exists\ u' \in R$: $u = u'.M$. Vì $v \in R[Z]$ nên $\exists\ v' \in R$: $v = v'.Z$. Vì $X = M \cap Z$ nên $X \subseteq M$ và $X \subseteq Z$, từ đó suy ra $u.X = u'.M.X = u'.X = v.X = v'.Z.X = v'.X$. Vì R thỏa PTH $X \to Y$ nên từ $u'.X = v'.X$ ta suy ra $u'.Y = v'.Y$ và do đó $u'.M = u'.XY = v'.XY = v'.M$. Từ đây suy ra $u*v = v' \in R$, *đpcm*.

5.2. Dùng kỹ thuật bảng để kiểm tra tính tổn thất của các phép tách sau:

a) $\rho = (U,F)$, $U = ABCD$,

$F = \{ A \to B, AC \to D \}$

$\rho = (AB, ACD)$.

$T \Rightarrow T^*$

	A	B	C	D
AB	a_1	a_2	b_{13}	b_{14}
ACD	a_1	b_{22} / a_2	a_3	a_4

Vì T^* chứa dòng thứ hai gồm toàn ký hiệu *phân biệt* nên phép tách đã cho là *không tổn thất*.

b) $\rho = (U,F)$, $U = ABCDE$,

$F = \{A \to C, B \to C, C \to D, DE \to C, CE \to A\}$

$\rho = (AD, AB, BE, CDE)$.

$T \Rightarrow T^*$

	A	B	C	D	E
AD	a_1	b_{12}	b_{13}/ a_3	a_4	b_{15}
AB	a_1	a_2	b_{23}/b_{13}/a_3	b_{24}/ a_4	b_{25}
BE	b_{31}	a_2	b_{33}/b_{13}/a_3	b_{34}/ a_4	a_5
CDE	b_{41} / b_{31}	b_{42}	a_3	a_4	a_5

Vì T^* *không chứa* một dòng *toàn ký hiệu phân biệt* nên phép tách đã cho là *tổn thất*.

5.3. Một lược đồ quan hệ ở dạng chuẩn 3 và có một khoá duy nhất có thể ở dạng chuẩn Boyce-Codd không? Vì sao?

Có.

Chứng minh: Giả sử LĐ p ở 3NF, có một khóa duy nhất K nhưng không ở BCNF. Vậy phải có một PTH không tầm thường X→A,

$A \notin X$ sao cho X không phải là siêu khóa. Ta có $K \rightarrow X$ vì K là khóa, $X \not\rightarrow K$ vì X không phải là siêu khóa. Nếu A không phải là thuộc tính khóa thì phụ thuộc bắc cầu $K \rightarrow X$, $X \not\rightarrow K$, $X \rightarrow A$, $A \notin X$ chứng tỏ LĐ đã cho không ở 3NF. Vậy A phải là thuộc tính khóa, tức là A nằm trong khóa duy nhất K. Xét tập $M = (K-A)X$. Vì $M \supseteq X$ nên theo tiên đề phản xạ Armstrong $M \rightarrow X$, mặt khác $X \rightarrow A$ theo giả thiết nên theo tiên đề bắc cầu Armstrong ta suy ra $M \rightarrow A$. Khi đó M^+ chứa A và do đó chứa K. Tức là M là siêu khóa không chứa A. Từ siêu khóa này ta có thể tìm được một khóa $K' \neq K$ (vì K chứa A mà K' không chứa A). Hai khóa này mâu thuẫn với điều kiện khóa K là duy nhất. Vậy giả thiết về sự tồn tại PTH không tầm thường $X \rightarrow A$, $A \notin X$ sao cho X không phải là siêu khóa là sai. LĐ phải ở dạng BCNF, đpcm.

5.4. Xác định và giải thích dạng chuẩn cao nhất của LĐQH sau:

p = (U, F); $U = ABCD$, $F = \{ A \rightarrow C, D \rightarrow B, C \rightarrow ABD \}$.

LĐ p có 2 khoá là A và C vì $A^+ = C^+ = ABCD = U$. Tập thuộc tính khoá là $U_k = AC$, tập thuộc tính không khoá là $U_0 = BD$. Ta có, $C \rightarrow D$ (?), $D \not\rightarrow C$ (?), $D \rightarrow B$. Đây là phụ thuộc bắc cầu của thuộc tính không khoá B vào khoá C. Vậy p không phải là 3NF và đương nhiên p không phải là BCNF(?). Vì hai khoá A và C đều chỉ có một thuộc tính nên các thuộc tính không khoá khác là B và D đều phụ thuộc đầy đủ vào hai khoá này. Vậy p là 2NF.

5.5. Chứng minh rằng một LĐQH ở dạng chuẩn 3 thì đồng thời ở dạng chuẩn 2.

Chứng minh: Nếu A là thuộc tính không khoá và A không phụ thuộc đầy đủ vào khoá K thì ta có $\exists M \subset K$: $M \rightarrow A$. Khi đó $M \not\rightarrow K$ (?), $A \notin M$

(?). Đây là một phụ thuộc bắc cầu, mâu thuẫn với giả thiết về 3NF. Vậy mọi thuộc tính không khoá phải phụ thuộc đầy đủ vào mọi khoá, tức là lược đồ phải là 2NF.

5.6. Chứng minh rằng một LĐQH ở dạng chuẩn BC thì đồng thời ở dạng chuẩn 3.

Chứng minh: Suy trực tiếp từ định nghĩa.

5.7. Chuẩn hoá 3NF CSDL **Thực tập**:

SV(SV#, HT, NS, QUE, HL)

DT(DT#, TDT, CN, KP)

SD(SV#, DT#, NTT, KM, KQ)

với các tập PTH sau:

F_{SV} = {SV# → HT, NS, QUE, HL}

F_{DT} = {DT# → TDT, CN, KP}

F_{SD} = {SV#, DT# → NTT, KQ ; NTT → KM}

Giải

Hai quan hệ SV và DT chỉ có 1 khoá đơn (1 thuộc tính) tương ứng là SV# và DT# và hai tập PTH tương ứng chỉ chứa 1 PTH nên hai quan hệ này ở BCNF(?).

Quan hệ SD có 1 khoá là K = SV#DT#. SD không ở 3NF vì có phụ thuộc bắc cầu của thuộc tính không khoá KM vào khoá K. Ta chuẩn hoá SD như sau:

1. *Tách*

   SV#, DT# → NTT

SV#, DT# → KQ

NTT → KM

2. *Tìm phủ tối tiểu*

   G = { SV#, DT# → NTT; SV#, DT# → KQ; NTT → KM }

3. *Gộp: ta thu lại được F*

4. *Kết quả: ρ = (SV#,DT#,NTT,KQ ; NTT, KM). Ta thu được hai quan hệ*

   SD(SV#,DT#,NTT,KQ) với khoá K=SV#,DT# và

   NK(NTT,KM) với khoá NTT

5.8. Chuẩn hoá 3NF LĐQH p = (U,F) sau:

U = MLTGSDP,

F = { M → T, GP → M, GT → P, MS → D, GS → P}

với ngữ nghĩa sau:

M: Môn học chuyên đề

L: Lớp chuyên đề

T; Thày - giáo viên phụ trách chuyên đề

G: Giờ học chuyên đề

S: Sinh viên theo học chuyên đề

D: Số đăng ký của sinh viên trong chuyên đề đó

P: Phòng học dành cho chuyên đề

M → T: Mỗi chuyên đề có một thày phụ trách

GP → M: Tại mỗi thời điểm, mỗi phòng học được dành cho không quá một môn

$GT \rightarrow P$: Tại mỗi thời điểm, mỗi thày dạy trong không quá một phòng học

$MS \rightarrow D$: Mỗi sinh viên tham gia chuyên đề nào thì được cấp một mã số ghi danh theo chuyên đề đó

$GS \rightarrow P$: Tại mỗi thời điểm, mỗi sinh viên có mặt trong không qúa một phòng học

1. Tìm một khoá: Tập thuộc tính có trong mọi khoá $K = LGS$. Ta có $K^+ = (LGS)^+ = LGSPMTD = U$ do đó lược đồ p có một khoá duy nhất là K. Tập các thuộc tính khoá: $U_k = LGS$, tập các thuộc tính không khoá: $U_0 = PMTD$.

2. Vì bộ phận thực sự của khoá K là GS và $GS \rightarrow P$, $P \in U_0$ nên p không ở $2NF$ và do đó không ở $3NF$.

3. Để ý rằng F đã ở dạng tối tiểu, ta có:

$\rho = (MT, GPM, GTP, MSD, GSP)$

Khoá $K = LGS$ không có trong thành phần nào của ρ nên ta thêm K vào để thu được kết quả sau:

$\rho = (MT, GPM, GTP, MSD, GSP, LGS)$

5.9. Chứng minh sự tương đương của hai định nghĩa *d* và *e* về *BCNF*.

LĐQH $p = (U,F)$ được gọi là

d) ở dạng chuẩn 3 mạnh hoặc Boyce-Codd (3*SNF*, *BCNF*) nếu p ở $1NF$ và mọi thuộc tính đều phụ thuộc trực tiếp vào mọi khoá,

e) ở dạng chuẩn 3 mạnh hoặc Boyce-Codd (3*SNF*, *BCNF*) nếu p ở 1NF và mọi PTH không tầm thường $X \rightarrow Y$ đều cho ta X là một siêu khóa.

Chứng minh

[$d \Rightarrow e$] Giả sử lược đồ p ở $1NF$ và mọi thuộc tính đều phụ thuộc trực tiếp vào mọi khoá và $X \rightarrow Y$ là một PTH không tầm thường. Ta phải chứng minh X là một siêu khoá. Vì PTH $X \rightarrow Y$ không tầm thường nên ta phải có $X \rightarrow A$ với $A \in Y \backslash X$. Nếu X không phải là siêu khoá thì với mọi khoá K ta có $X \not\rightarrow K$, $K \rightarrow X$, $X \rightarrow A$, $A \notin X$, tức là A phụ thuộc bắc cầu vào khoá K, mâu thuẫn với định nghĩa d. Vậy X phải là siêu khoá, tức là ta có e, đpcm.

[$e \Rightarrow d$] Giả sử lược đồ p ở $1NF$ và mỗi PTH không tầm thường $X \rightarrow Y$ đều cho ta X là một siêu khoá. Cho khoá K của p và một thuộc tính A trong U. Giả sử A phụ thuộc bắc cầu vào K, tức là $\exists X \subseteq U$: $K \rightarrow X$, $X \not\rightarrow K$, $X \rightarrow A$, $A \notin X$. Như vậy $X \rightarrow A$ là PTH không tầm thường nên ta phải có X là một siêu khoá của p và do đó $X \rightarrow K$, mâu thuẫn với giả thiết về phụ thuộc bắc cầu. Vậy mọi thuộc tính A phải phụ thuộc trực tiếp vào mọi khoá của p, tức là ta có d, đpcm.

BÀI GIẢI
CÁC ĐỀ THI

BÀI GIẢI

ĐỀ THI TUYỂN CAO HỌC ĐẠI HỌC THÁI NGUYÊN NĂM 2002

Môn: Cơ sở công nghệ thông tin

Câu 5.

a.1) Định nghĩa quan hệ:

Cho tập hữu hạn $U = \{A_1, A_2, \dots, A_n\}$ khác trống ($n \geq 1$). Các phần tử của U được gọi là *thuộc tính*. Ứng với mỗi thuộc tính $A_i \in U$, $i = 1,2, \dots, n$ có một tập không rỗng $dom(A_i)$ được gọi là *miền trị* (miền biến thiên) của thuộc tính A_i. Ta đặt $D = \bigcup_{A_i \in U} dom(A_i)$.

Một *quan hệ* R với các thuộc tính $U = \{A_1, A_2, \dots, A_n\}$, ký hiệu là $R(U)$, là một tập các ánh xạ $t: U \to D$ sao cho với mỗi $A_i \in U$ ta có $t(A_i) \in dom(A_i)$. Mỗi ánh xạ được gọi là *một bộ* của quan hệ R.

a.2) Định nghĩa phụ thuộc hàm:

Cho tập thuộc tính U. Một phụ thuộc hàm (PTH) trên U là công thức dạng $f: X \to Y$; $X, Y \subseteq U$.

Cho quan hệ $R(U)$ và một PTH $f: X \to Y$ trên U. Ta nói quan hệ R thoả PTH f và viết $R(f)$, nếu hai bộ tuỳ ý trong R giống nhau trên X thì chúng cũng giống nhau trên Y,

$(\forall\, u,v \in R)$: $(u.X = v.X) \Rightarrow (u.Y = v.Y)$

b.1) Phát biểu hệ tiên đề Armstrong:

Cho tập PTH F trên tập thuộc tính U. Bao đóng của F, ký hiệu F^+ là tập nhỏ nhất các PTH trên U chứa F và thoả các tính chất F1-F3 của hệ tiên đề Armstrong $A°$ sau đây:

F1. Tính phản xạ: Nếu $X \supseteq Y$ thì $X \rightarrow Y \in F^+$

F2. Tính gia tăng: Nếu $X \rightarrow Y \in F^+$ thì $XZ \rightarrow YZ \in F^+$

F3. Tính bắc cầu: Nếu $X \rightarrow Y \in F^+$ và $Y \rightarrow Z \in F^+$ thì $X \rightarrow Z \in F^+$

Nếu $f \in F^+$ ta nói tập PTH F dẫn ra được PTH f theo tiên đề Armstrong.

Cho tập PTH F trên tập thuộc tính U và f là một PTH trên U. Ta nói PTH f được suy dẫn theo quan hệ từ tập PTH F, nếu mọi quan hệ $R(U)$ thoả F thì R cũng thoả f. Ta ký hiệu F^* là tập toàn bộ các PTH f được suy dẫn theo quan hệ từ tập PTH F.

b.2) Phát biểu tính đúng của hệ tiên đề Armstrong:

Định lý: *Cho tập PTH F trên tập thuộc tính U và một PTH f trên U. Nếu f được dẫn từ F theo tiên đề Armstrong thì f cũng được dẫn từ F theo quan hệ, tức là* $F^+ \subseteq F^*$.

c.1) Định nghĩa lược đồ quan hệ: Lược đồ quan hệ (LĐQH) là một cặp $p = (U,F)$, trong đó U là tập hữu hạn các thuộc tính, F là tập các phụ thuộc hàm trên U.

c.2) Định nghĩa khoá của lược đồ quan hệ: Cho LĐQH $p = (U,F)$. Tập thuộc tính $K \subseteq U$ được gọi là khoá của LĐ p nếu

(i) $K^+ = U$

(ii) $\forall A \in K: (K - \{A\})^+ \neq U$

d.1) Phát biểu bài toán thành viên đối với lược đồ quan hệ:

Cho tập thuộc tính U, một tập các PTH F trên U và một PTH f: $X \rightarrow Y$ trên U. Hỏi rằng $X \rightarrow Y \in F^+$ hay không ?

d.2) Kết quả liên quan tới bài toán thành viên:

Định lý : *$X \rightarrow Y \in F^+$ khi và chỉ khi $Y \subseteq X^+$.*

Vì đã có thuật toán tìm bao đóng của tập thuộc tính với độ phức tạp thời gian đa thức nên định lý trên cho ta thuật toán giải bài toán thành viên với độ phức tạp thời gian đa thức.

Câu 6.

a.1) Thuật toán tính bao đóng của một tập thuộc tính trong một lược đồ quan hệ:

Ý tưởng: Cho tập PTH F trên tập thuộc tính U và một tập con các thuộc tính X trong U. Để xác định bao đóng của tập thuộc tính X, X^+ ta xuất phát từ tập X và bổ sung dần cho X các thuộc tính thuộc vế phải R của các PTH $L \rightarrow R \in F$ thỏa điều kiện $L \subseteq X$. Thuật toán sẽ dừng khi không thể bổ sung thêm thuộc tính nào cho X.

```
Algorithm Closure
Format: C = X+
Input:      - LĐQH p =(U,F)
            - Tập thuộc tính X ⊆ U
Output: - Y = X+ = { A ∈ U | X → A }
Method
      Y:=X;
```

```
        repeat
                Z:=Y;
                for each FD L→R in F do
                        if L ⊆ Y then
                                Y:=Y ∪ R;
                        endif;
                endfor;
        until Y=Z;
        return Y;
    end Closure;
```

b) Thuật toán tìm một khoá của lược đồ quan hệ

Tư tưởng: Xuất phát từ một siêu khóa *K*, loại dần các thuộc tính trong *K*, bảo toàn bất biến $K^+ = U$.

```
Algorithm Key
    Format: Key(U,F)
    Input:      - Tập thuộc tính U
                - Tập PTH F
    Output: - Khóa K ⊆ U thỏa
      • K+ = U
      • ∀ A ∈ K: (K-{A})+ ≠ U
    Method
        K:=U;
```

```
        for each attribute A in U do
            if (K-{A})+ = U then
                K := K-{A}
            endif;
    endfor;
    return K;
end Key;
```

c) Giả thiết:

LĐQH s = (*R*,*F*);

R = *abcdefg*,

F = { *abc* → *de*, *bcd* → *g*, *abf* → *eg*, *ce* → *fg* }

Tìm một khoá của lược đồ *s*.

LĐQH *s* có khóa *K* = *abc*, vì:

$(abc)^+ = abcdefg = U$ và $(bc)^+ = bc \neq U$, $(ac)^+ = ac \neq U$, $(ab)^+ = ab \neq U$.

BÀI GIẢI

ĐỀ THI TUYỂN NGHIÊN CỨU SINH VIỆN CÔNG NGHỆ THÔNG TIN NĂM 2002

Đề số 1

Câu 1.

1.a) Định nghĩa lược đồ quan hệ:

Lược đồ quan hệ (LĐQH) là một cặp $p = (U,F)$, trong đó U là tập hữu hạn các thuộc tính, F là tập các phụ thuộc hàm trên U.

1.b) Định nghĩa bao đóng của tập thuộc tính trong LĐQH:

Cho tập PTH F trên tập thuộc tính U và một tập con các thuộc tính X trong U. Bao đóng của tập thuộc tính X, ký hiệu X^+ là tập thuộc tính $X^+ = \{ A \in U \mid X \rightarrow A \in F^+ \}$

1.c) Thuật toán tìm bao đóng của tập thuộc tính trong LĐQH:

Ý tưởng: Cho tập PTH F trên tập thuộc tính U và một tập con các thuộc tính X trong U. Để xác định bao đóng của tập thuộc tính X, X^+ ta xuất phát từ tập X và bổ sung dần cho X các thuộc tính thuộc vế phải R của các PTH $L \rightarrow R \in F$ thỏa điều kiện $L \subseteq X$. Thuật toán sẽ dừng khi không thể bổ sung thêm thuộc tính nào cho X.

```
Algorithm Closure
```

```
Format: C = X+
Input:        - LĐQH p =(U,F)
              - Tập thuộc tính X ⊆ U
Output: - Y = X+ = { A ∈ U  | X → A ∈ F+ }
Method
      Y:=X;
    repeat
            Z:=Y;
            for each FD L→R in F do
                  if L ⊆ Y then
                        Y:=Y ∪ R;
                  endif;
            endfor;
      until Y=Z;
      return Y;
end Closure;
```

Câu 2.

Giả thiết:

LĐQH $p = (U, F)$;

$U = ABCDEGH$;

$F = \{CD \to H,\ E \to B,\ D \to G,\ BH \to E,\ CH \to DG,\ C \to A\}$.

a) Tìm tập M là giao của toàn bộ các khoá của p:

$$M = U - \bigcup_{L \to R \in F} (R - L) = C$$

b) Cho biết *p* có đúng 1 khoá hay không?

Vì $C^+ = CA \neq U$ nên LĐQH p có hơn 1 khóa.

c) Tập *ABD* có phải là khoá của *p* không? Vì sao?

ABD không phải là khóa của p vì nó không chứa C là phần có mặt trong mọi khóa.

d) Tập *CH* có phải là khoá của *p* không? Vì sao?

Tập *CH* không phải là khoá của *p* vì $(CH)^+ = ACDGH \neq U$.

e) Tính $Z = (X^+ Y)^+ \cap (K^+ - Y)$ *biết* $X = CD$, $Y = CH$, *K* là một siêu khoá của p.

Vì K là siêu khóa nên $K^+ = U$, do đó $K^+ - Y = ABCDEGH - CH = ABDEG$. Ta lại biết $(X^+ Y)^+ = (X Y)^+ = (CDH)^+ = ACDGH$, nên

$Z = ACDGH \cap ABDEG = ADG$.

Câu 3. Phép phân tách một LĐQH:

3.a) Cho lược đồ quan hệ p = (U,F). Một phép tách trên tập thuộc tính U là một họ các tập con của U $\rho = (X_1, X_2, ..., X_k)$ thỏa tính chất:

$$\bigcup_{i=1}^{k} X_i = U.$$

Phép tách $\rho = (X_1, X_2, ..., X_k)$ trên tập thuộc tính U được gọi là *không tổn thất* (hoặc *không mất thông tin)* đối với tập PTH F nếu

$\forall R(U) \in SAT(F)$: $R[X_1] * R[X_2] * ... * R[X_k] = R$. Ngược lại, nếu không tồn tại đẳng thức thì ta gọi ρ là phép tách *tổn thất*.

Kiểm tra tính tổn thất của phép tách bằng kỹ thuật bảng

Input

- LĐQH p = (U,F)
- Phép tách $\rho = (X_1, X_2,...,X_k)$

Output

- True, nếu ρ là một phép tách không tổn thất
- False, ngoài ra.

Method

1. *Khởi trị:* Lập bảng T với các cột là các thuộc tính trong U và k dòng, mỗi dòng ứng với một thành phần của X_i trong ρ: Dòng i chứa các ký hiệu phân biệt (KHPB) a_j ứng với các thuộc tính A_j trong X_i và các ký hiệu không phân biệt (KHKPB) b_{ij} ứng với các thuộc tính A_j trong $U-X_i$. Chú ý rằng mọi KHPB trong cột j của T là giống nhau và bằng a_j còn mọi KHKPB trong bảng T lúc đầu là khác nhau.

2. *Sửa bảng:* Lặp đến khi bảng T không còn thay đổi:

2.1. Vận dụng các F-luật để biến đổi bảng như sau:

Với mỗi PTH $L \rightarrow R$ trong F, nếu trong bảng T có chứa hai dòng u và v giống nhau trên L thì sửa các ký hiệu của chúng cho giống nhau trên mọi cột A trong bảng T như sau:

a) nếu u.A = v.A: không sửa,

b) nếu chỉ một trong hai ký hiệu u.A hoặc v.A là KHPB thì sửa ký mọi xuất hiện trong bảng của KHKPB thành KHPB đó,

c) nếu cả hai ký hiệu u.A và v.A đều là KHKPB thì sửa mọi xuất hiện trong bảng của ký hiệu có chỉ số thứ nhất lớn hơn thành ký hiệu thứ hai.

3. *Kết luận:*

Gọi bảng kết quả là T*.

Nếu T* chứa một dòng toàn KHPB thì return True nếu không return False.

end.

3.c) Cho LĐQH *p* như trong câu 2. Vận dụng thuật toán kiểm tra phép phân tách *w* có mất thông tin hay không?

Giả thiết

$p = (U, F)$,

$U = ABCDEGH$,

$F = \{ CD \rightarrow H,\ E \rightarrow B,\ D \rightarrow G,\ BH \rightarrow E,\ CH \rightarrow DG,\ C \rightarrow A \}$.

$w = [ABCDE, BCH, CDEGH]$

$T \Rightarrow T^*$

	A	B	C	D	E	G	H
ABCDE	a_1	a_2	a_3	a_4	a_5	b_{16}/a_6	b_{17}/a_7
BCH	b_{21}/a_1	a_2	a_3	b_{24}/a_4	b_{25}/a_5	b_{26}/a_6	a_7
CDEGH	b_{31}/a_1	b_{32}/a_2	a_3	a_4	a_5	a_6	a_7

Do T có chứa dòng toàn ký hiệu phân biệt nên phép tách w đã cho là không tổn thất.

Câu 4.

4.a) Định nghĩa các dạng chuẩn *2NF*, *3NF* và *BCNF* của LĐQH:

LĐQH p = (U,F) được gọi là

- ở dạng chuẩn 1 (1NF) nếu mọi thuộc tính trong U đều không phải là thuộc tính phức hợp,
- ở dạng chuẩn 2 (2NF) nếu p ở 1NF và mọi thuộc tính không khoá đều phụ thuộc đầy đủ vào mọi khoá,
- ở dạng chuẩn 3 (3NF) nếu p ở 1NF và mọi thuộc tính không khoá đều phụ thuộc trực tiếp vào mọi khoá,
- ở dạng chuẩn 3 mạnh hoặc Boyce-Codd (3SNF, BCNF) nếu p ở 1NF và mọi PTH không tầm thường $X \rightarrow Y$ đều cho ta X là một siêu khóa.

4.b) Sơ đồ tương quan giữa các dạng chuẩn:

BCNF $\Rightarrow$ 3NF $\Rightarrow$ 2NF

4.c) Xác định dạng chuẩn cao nhất của LĐQH *h* sau đây:

$h = (U, F);\ U = ABCD,\ F = \{ D \rightarrow B,\ C \rightarrow A,\ B \rightarrow ACD \}$

Giải thích vì sao?

Lược đồ h có 2 khóa $K_1 = D;\ K_2 = B$ vì $D^+ = ABCD$ và $B^+ = ABCD$.

Vì có PTH $C \rightarrow A$ mà *C* không phải là siêu khóa nên h không thể ở dạng chuẩn BCNF.

Tập thuộc tính khóa là $U_K = BD$; Tập thuộc tính không khóa là $U_0 = AC$;

Ta có:

$B \rightarrow C$ (vì *B* là khóa),

$C \not\rightarrow B$ (vì *C* không là khóa)

$C \rightarrow A$ (giả thiết)

và A là thuộc tính không khóa. Như vậy thuộc tính không khóa *A* phụ thuộc bắc cầu vào khóa *B* nên *h* không ở dạng chuẩn 3NF.

Hai thuộc tính không khóa *A* và *C* đều phụ thuộc đầy đủ vào các khóa *B'* và *D* do đó lược đồ *h ở dạng chuẩn 2NF.*

BÀI GIẢI

ĐỀ THI TUYỂN NGHIÊN CỨU SINH
VIỆN CÔNG NGHỆ THÔNG TIN
NĂM 2002

Đề số 2

Câu 1.

1.a) Định nghĩa PTH: Cho tập thuộc tính U. Một phụ thuộc hàm (PTH) trên U là công thức dạng f: $X \rightarrow Y$; $X, Y \subseteq U$.

Cho quan hệ R(U) và một PTH f: $X \rightarrow Y$ trên U. Ta nói quan hệ R thoả PTH f và viết R(f), nếu hai bộ tuỳ ý trong R giống nhau trên X thì chúng cũng giống nhau trên Y,

$$(\forall\ u,v \in R): (u.X = v.X) \Rightarrow (u.Y = v.Y)$$

1.b) Định nghĩa LĐQH: Lược đồ quan hệ (LĐQH) là một cặp p = (U,F), trong đó U là tập hữu hạn các thuộc tính, F là tập các phụ thuộc hàm trên U.

1.c) Phát biểu bài toán thành viên trên LĐQH: Cho tập thuộc tính U, một tập các PTH F trên U và một PTH f: $X \rightarrow Y$ trên U. Hỏi rằng

$X \rightarrow Y \in F^+$ hay không ?

1.d) Điều kiện cần và đủ liên quan đến bài toán thành viên:

Định lý: $X \rightarrow Y \in F^+$ khi và chỉ khi $Y \subseteq X^+$.

Câu 2.

2.a) Định nghĩa khoá của LĐQH: Cho LĐQH p = (U,F). Tập thuộc tính $K \subseteq U$ được gọi là khoá của LĐ p nếu

(i) $K^+ = U$

(ii) $\forall A \in K: (K - \{A\})^+ \neq U$

2.b) Thuật toán tìm 1 khoá của LĐQH:

Tư tưởng: Xuất phát từ một siêu khóa K, loại dần các thuộc tính trong K, bảo toàn bất biến $K^+ = U$.

```
Algorithm Key
    Format: Key(U,F)
    Input:      - Tập thuộc tính U
                - Tập PTH F
    Output: - Khóa K ⊆ U thỏa
      • K+=U
      • ∀A ∈ K: (K-{A})+ ≠ U
    Method
        K:=U;
      for each attribute A in U do
          if (K-{A})+ = U then
              K := K-{A}
          endif;
      endfor;
      return K;
    end Key;
```

Câu 3.

3.a) Định nghĩa thuộc tính khoá (thuộc tính cơ bản hay nguyên thuỷ), thuộc tính không khoá (thuộc tính thứ cấp): Cho LĐQH p = (U,F). Thuộc tính A trong U được gọi là thuộc tính *khóa* nếu A có trong một khóa của p. A được gọi là thuộc tính *không khóa* nếu A không có trong bất kỳ khóa nào của p.

3.b)

Giả thiết:

LĐQH $s = (U, F)$

$U = ABCD$,

$F = \{AD \rightarrow BC,\ B \rightarrow A,\ D \rightarrow C\}$.

a) Tìm các khoá của *s*: Lược đồ s có 2 khóa:

$K_1 = BD$, vì $(BD)^+ = U$; $B^+ = AB \neq U$; $D^+ = CD \neq U$.

$K_2 = AD$; vì $(AD)^+ = U$; $A^+ = A \neq U$ và $D^+ \neq U$;

b) Cho biết *C* có phải là thuộc tính khoá hay không? *C* không phải là thuộc tính khóa của s vì *C* không có trong khóa nào.

Câu 4.

4.a) Định nghĩa các dạng chuẩn *2NF*, *3NF* và *BCNF* của LĐQH: LĐQH p = (U,F) được gọi là

- ở *dạng chuẩn 1* (1NF) nếu mọi thuộc tính trong U đều không phải là thuộc tính phức hợp,
- ở *dạng chuẩn 2* (2NF) nếu p ở 1NF và mọi thuộc tính không khoá đều phụ thuộc đầy đủ vào mọi khoá,

- *ở dạng chuẩn 3* (3NF) nếu p ở 1NF và mọi thuộc tính không khoá đều phụ thuộc trực tiếp vào mọi khoá,
- *ở dạng chuẩn 3 mạnh* hoặc *Boyce-Codd* (3SNF, BCNF) nếu p ở 1NF và mọi PTH không tầm thường $X \rightarrow Y$ đều cho ta X là một siêu khóa.

4.b) Sơ đồ tương quan giữa các dạng chuẩn:

$$BCNF \Rightarrow 3NF \Rightarrow 2NF$$

4.c) Xác định dạng chuẩn cao nhất của LĐQH *h* sau:

$h = (U, F); U = ABCD, F = \{ CD \rightarrow B, A \rightarrow C, B \rightarrow ACD \}$

Giải thích vì sao?

Lược đồ *h* có 3 khóa: $K_1 = B$, vì $B^+ = U$;

$K_2 = CD$, vì $(CD)^+ = U$, $C^+ = C \neq U$, $D^+ = D \neq U$.

$K_3 = AD$, vì $(AD)^+ = U$, $A^+ = AC \neq U$, $D^+ = D \neq U$.

Tập thuộc tính khóa là $U_K = ABCD = U$, vậy *h* không có thuộc tính không khóa.

Ta có PTH $A \rightarrow C$ mà *A* không phải là siêu khóa nên *h* không thể ở dạng chuẩn BCNF.

Do *h* không có thuộc tính không khóa nên không có phụ thuộc bắc cầu của thuộc tính không khóa vào các khóa. Vậy *h* ở *dạng chuẩn 3NF*.

Câu 5.

Giả thiết:

$p = (U, F); U = ABCDEHG;$

$F = \{ DE \rightarrow G, H \rightarrow C, E \rightarrow A, CG \rightarrow H, DG \rightarrow EA, D \rightarrow B \}$..

a. Tìm tập *M* là giao của toàn bộ các khoá của *p*. Cho biết *p* có đúng 1 khoá hay không?

M = U - ABCGEH = D. $M^+ = D^+ = BD \neq U$ nên p có hơn 1 khóa.

b. Tìm 1 khoá của *p*: *K = DEH, vì* $(DEH)^+$ *= DEHGCAB,*

$(DE)^+ = DEGAB \neq U$, $(DH)^+ = DHCB \neq U$, $(EH)^+ = EHAC \neq U$.

c. Tập *BCE* có phải là khoá của *p* không? Vì sao?

Không, vì BCE không chứa D là giao các khóa.

d. Hãy thêm hoặc bớt 1 phụ thuộc hàm cho *F* để LĐQH có đúng 1 khoá. *Thêm chẳng hạn PTH D→ABCEGH. Khi đó D vẫn là giao các khóa và* $D^+ = U$ *nên p có đúng 1 khóa.*

BÀI GIẢI

ĐỀ THI TUYỂN NGHIÊN CỨU SINH
ĐẠI HỌC BÁCH KHOA HÀ NỘI
NĂM 2002

Bài 1.

1a) Định nghĩa lược đồ quan hệ:

LĐQH là một cặp $p = (U,F)$ trong đó U là tập hữu hạn các thuộc tính, F là tập PTH trên U.

1b) Định nghĩa khoá của lược đồ quan hệ: Cho LĐQH $p = (U,F)$. Khóa của LĐQH là tập con thuộc tính K của U thỏa

(i) $K^+ = U$

(ii) $\forall A \in K$: $(K-\{A\})^+ \neq U$.

1c) Thuật toán tìm một khoá của lược đồ quan hệ:

Tư tưởng: Xuất phát từ một siêu khóa K, loại dần các thuộc tính trong K, bảo toàn bất biến $K^+ = U$.

```
Algorithm Key
    Format: Key(U,F)
    Input:      - Tập thuộc tính U
                - Tập PTH F
    Output:     - Khóa K ⊆ U thỏa
```

- $K^+=U$
- $\forall A \in K: (K-\{A\})^+ \neq U$

```
Method
        K:=U;
      for each attribute A in U do
            if (K-{A})+ = U then
                K := K-{A}
            endif;
      endfor;
      return K;
    end Key;
```

Bài 2. Giả thiết:

$p = (U,F)$; $U = ABCDEH$

$F = \{BC \to E,\ D \to A,\ C \to A,\ AE \to D,\ BE \to CH\}$.

a) Tìm một khoá K của lược đồ p:

LĐ p có khóa $K = BC$ vì

$(BC)^+ = BCEADH = U$, $B^+ = B \neq U$, $C^+ = CA \neq U$.

b) Ngoài khoá K, lược đồ p còn khoá nào khác không? Vì sao?

Giao của các khóa: $M = U - EADCH = B$; $M^+ = B^+ = B \neq U$. Vậy LĐ có hơn 1 khóa.

c) Tập BCH có phải là khoá của p không? Vì sao? BCH không phải là khóa của p vì BCH chứa thực sự khóa $K = BC$ tìm được ở câu a.

d) Tập BD có phải là khoá của p không? Vì sao?

Vì $(BD)^+ = BDA \neq U$ nên BD không phải là khóa của p.

e) Tính $Z = (X^+ \cup Y)^+ \cap K^+ - (X \cup Y)$ với $X = AB$, $Y = D$, K là một siêu khoá của p.

Vì K là siêu khóa nên $K^+ = U$, mặt khác $(X^+ \cup Y)^+ = (X \cup Y)^+$, do đó $Z = (X \cup Y)^+ - (X \cup Y) = (ABD)^+ - ABD = ABD - ABD = \varnothing$.

f) Hãy thêm cho F một phụ thuộc hàm để p có đúng một khoá. Giải thích cách làm:

Vì giao các khóa là B nên ta có thể thêm PTH $B \rightarrow ACDEH$. Khi đó giao các khóa vẫn là B và ta có $B^+ = U$ nên LĐ có đúng 1 khóa.

Bài 3. Một lược đồ quan hệ ở dạng chuẩn 3 và có một khoá duy nhất có thể ở dạng chuẩn Boyce-Codd không? Vì sao? *Có.*

Chứng minh: Giả sử LĐ p ở 3NF, có một khóa duy nhất K nhưng không ở BCNF. Vậy phải có một PTH không tầm thường $X \rightarrow A$, $A \notin X$ sao cho X không phải là siêu khóa. Ta có $K \rightarrow X$ vì K là khóa, $X \not\rightarrow K$ vì X không phải là siêu khóa. Nếu A không phải là thuộc tính khóa thì phụ thuộc bắc cầu $K \rightarrow X$, $X \not\rightarrow K$, $X \rightarrow A$, $A \notin X$ chứng tỏ LĐ đã cho không ở 3NF. Vậy A phải là thuộc tính khóa, tức là A nằm trong khóa duy nhất K. Xét tập $M = (K-A)X$. Vì $M \supseteq X$ nên theo tiên đề phản xạ Armstrong $M \rightarrow X$, mặt khác $X \rightarrow A$ theo giả thiết, nên theo tiên đề bắc cầu Armstrong ta suy ra $M \rightarrow A$. Khi đó M^+ chứa A và do đó chứa K. Tức là M là siêu khóa không chứa A. Từ siêu khóa này ta có thể tìm được một khóa $K' \neq K$ (vì K chứa A mà K' không chứa A). Hai khóa này mâu thuẫn với điều kiện khóa K là duy nhất. Vậy giả thiết về sự tồn tại PTH không tầm thường $X \rightarrow A$, $A \notin X$ sao cho X không phải là siêu khóa là sai. LĐ phải ở dạng BCNF, đpcm.

BÀI GIẢI

ĐỀ THI TUYỂN CAO HỌC NGÀNH GIS NĂM 2003
ĐẠI HỌC BÁCH KHOA TP HỒ CHÍ MINH
Môn: Cơ sở dữ liệu, đề số 1

Thời gian 120 phút, không tham khảo tài liệu

Trong đề thi, tập các thuộc tính được viết dưới dạng xâu ký tự, ký hiệu XY được hiểu là phép hợp 2 tập X và Y, XY = X∪Y. Các từ viết tắt: LĐQH: lược đồ quan hệ, PTH: phụ thuộc hàm.

Câu 1.

a) Định nghĩa phép trừ hai quan hệ:

Phép trừ (theo lý thuyết tập hợp hoặc *lấy phần riêng*) hai quan hệ tương thích $R(U)$ và $S(U)$, ký hiệu R-S, cho ta quan hệ chứa các bộ của quan hệ R không có trong quan hệ S,

$$P(U) = R\text{-}S = \{ t \mid t \in R,\ t \notin S \}$$

b) Thuật toán xác định phép trừ hai quan hệ

```
Algorithm Substraction
Format: P = R-S
Input:      - Quan hệ R(U)
            - Quan hệ S(U)
```

```
Output: - Quan hệ R-S={t|t∈R,t∉S}
Method
    Create(P,Attr(R));
      for each tuple u in R
            with u not_in S do
                  add u to P;
      endfor;
      return P;
end  Substraction;
```

Câu 2.

a) *Định nghĩa LĐQH: Lược đồ quan hệ (LĐQH) là một cặp p = (U,F), trong đó U là tập hữu hạn các thuộc tính, F là tập các phụ thuộc hàm trên U.*

b) *Định nghĩa bao đóng của tập thuộc tính trong LĐQH: Cho LĐQH p = (U,F), trong đó U là tập thuộc tính, F là tập PTH trên U. Bao đóng của tập thuộc tính X, ký hiệu X^+ là tập thuộc tính*

$$X^+ = \{A \in U \mid X \rightarrow A \in F^+\}$$

c) *Thuật toán tìm bao đóng của tập thuộc tính trong LĐQH:*

Ý tưởng: *Cho tập PTH F trên tập thuộc tính U và một tập con các thuộc tính X trong U. Để xác định bao đóng của tập thuộc tính X, X^+ ta xuất phát từ tập X và bổ sung dần cho X các thuộc tính thuộc vế phải R của các PTH $L \rightarrow R \in F$ thỏa điều kiện $L \subseteq X$. Thuật toán sẽ dừng khi không thể bổ sung thêm thuộc tính nào cho X.*

```
Algorithm Closure
Format: C = X+
Input:        - LĐQH p =(U,F)
              - Tập thuộc tính X ⊆ U
Output: - Y = X+ = {A∈U | X→A∈F+}
Method
      Y:=X;
    repeat
            Z:=Y;
            for each FD L→R in F do
                  if L ⊆ Y then
                        Y:=Y ∪ R;
                  endif;
            endfor;
        until Y=Z;
        return Y;
  end Closure;
```

Câu 3.

Giả thiết:

LĐQH p = (U, F), U = ABCDE,

F = { DE → A, B → C, E → AD }.

a) Tìm một khoá của lược đồ p. Lược đồ p có khoá K=BE, vì $(BE)^+=U$ và $B^+ = BC \neq U$; $E^+ = ADE \neq U$.

b) Tập BCE có phải là khoá của p không? Vì sao?

Không, vì BCE chứa BE, mà BE là khoá.

c) Tập AD có phải là khoá của p không? Vì sao?

Không, vì $(AD)^+ = AD \neq U$.

d) Lược đồ p còn khoá nào nữa không? Vì sao?

Ta tìm tập *M* là giao của toàn bộ các khoá của *p*:

$$M = U - \bigcup_{L \to R \in F} (R - L).$$

M = ABCDE - ACD = BE = K do đó LĐQH p chỉ có một khoá duy nhất.

e) Tính $Z = (X^+Y)^+ \cap (K^+ - Y)$ *biết X = DE, Y = AD, K là một siêu khoá của p.*

Nhận xét: Vì K là siêu khoá của p nên $K^+ = U$. *Mặt khác, theo công thức* $(X^+Y)^+ = (XY)^+$ *nên ta có*

$Z = (XY)^+ \cap (U - Y) = (DEA)^+ \cap BCE = DEA \cap BCE = E$.

f) Có thể thêm vào F một PTH để thu được lược đồ có đúng 2 khoá không. Giải thích cách làm?

Thêm, chẳng hạn PTH $C \to B$. *Khi đó ngoài khoá BE lược đồ có thêm khoá CE, vì* $C \to B$. *Dễ thấy có thể tách LĐQH p thành 2 LĐQH thành phần độc lập nhau là* p_1=(ADE, {DE→A, E→AD}) với khoá K_1= E *và* p_2 =(BC, {B→C}) với khoá K_2 = B. Khi thêm *PTH* $C \to B$ *cho LĐQH* p_2 *ta có thêm khoá* $K_3 = C$. *Hợp nhất 2 lược đồ này lại ta sẽ được đúng 2 khoá là*

$K_1K_2 = EB$ *và* $K_1K_3 = EC$.

BÀI GIẢI

ĐỀ THI TUYỂN CAO HỌC NGÀNH GIS NĂM 2003
ĐẠI HỌC BÁCH KHOA TP HỒ CHÍ MINH
Môn: Cơ sở dữ liệu, đề số 2

Thời gian 120 phút, không tham khảo tài liệu

Trong đề thi, tập các thuộc tính được viết dưới dạng xâu ký tự, ký hiệu XY được hiểu là phép hợp 2 tập X và Y, XY = X∪Y. Các từ viết tắt: LĐQH: lược đồ quan hệ, PTH: phụ thuộc hàm.

Câu 1.

a) Định nghĩa phép toán kết nối tự nhiên hai quan hệ: Phép kết nối (tự nhiên) hai quan hệ $R(U)$ và $S(V)$, ký hiệu $R*S$, cho ta quan hệ chứa các bộ được dán từ các bộ u của quan hệ R với mỗi bộ v của quan hệ S sao cho các trị trên miền thuộc tính chung (nếu có) của hai bộ này giống nhau.

$$P(UV) = R*S = \{ u*v \mid u \in R, v \in S, u.M = v.M, M = U \cap V \}$$

b) Thuật toán xác định phép toán kết nối tự nhiên hai quan hệ:

```
Algorithm Join
Format: P = R*S
Input:      - Quan hệ R(U)
            - Quan hệ S(V)
Output: - Quan hệ
```

```
        R*S={u*v|u∈R, v∈S, u.M=v.M, M=U∩V}
    Method
        X = Attr(R) ∪ Attr(S);
        M = Attr(R) ∩ Attr(S);
      Create(P,X);
        for each tuple u in R do
              for each tuple v in S do
              if u.M = v.M then
                        add u*v to P ;
                    endif;
              endfor;
        endfor;
        return P;
    end Join;
```

Câu 2.

a) *Định nghĩa LĐQH: Lược đồ quan hệ (LĐQH) là một cặp p = (U,F), trong đó U là tập hữu hạn các thuộc tính, F là tập các phụ thuộc hàm trên U.*

b) *Định nghĩa khoá của lược đồ quan hệ: Cho LĐQH p = (U,F). Khóa của LĐQH là tập con thuộc tính K của U thỏa*

(i) $K^+ = U$

(ii) $\forall A \in K: (K-\{A\})^+ \neq U$.

c) *Thuật toán tìm một khoá của lược đồ quan hệ:*

Tư tưởng: *Xuất phát từ một siêu khóa K, loại dần các thuộc tính trong K, bảo toàn bất biến $K^+ = U$.*

```
Algorithm Key
        Format: Key(U,F)
        Input:        - Tập thuộc tính U
                        Tập PTH F
        Output: - Khóa K ⊆ U thỏa
           •  K+ = U
           •  ∀ A∈ K: (K-{A})+ ≠ U
        Method
              K:=U;
            for each attribute A in U do
                  if (K-{A})+ = U then
                      K := K-{A}
                  endif;
           endfor;
           return K;
         end Key;
```

Câu 3.

Giả thiết:

LĐQH p = (U, F); U = ABCDE;

F = { EA → B, C → D, A → BE }.

a) Tìm một khoá của lược đồ p:

Ta tìm tập *M* là giao của toàn bộ các khoá của *p:*

$$M = U - \bigcup_{L \to R \in F} (R - L)$$

M = ABCDE - BDE = AC, vì $(AC)^+ = U$ nên p có một khoá duy nhất là M = AC.

b) Tập CDA có phải là khoá của p không? Vì sao?

Không, vì CDA chứa khoá AC.

c) Tập BE có phải là khoá của p không? Vì sao?

Không, vì p chỉ có một khoá duy nhất là AC.

d) Lược đồ p còn khoá nào nữa không? Vì sao?

LĐQH p chỉ có một khoá duy nhất M=AC theo lý do trong câu a).

e) Tính $Z = (X^+ Y)^+ \cap K^+$ biết X = AE, Y = BE, K là một khoá của p.

Nhận xét: Vì K là khoá của p nên $K^+ = U$. Mặt khác, theo công thức $(X^+Y)^+ = (XY)^+$ nên ta có

$Z = (XY)^+ \cap U = (XY)^+ = (ABE)^+ = ABE$.

f) Có thể thêm vào F một PTH để thu được lược đồ có đúng 2 khoá không. Giải thích cách làm?

Thêm, chẳng hạn PTH $D \to C$ Khi đó ngoài khoá AC lược đồ có thêm khoá AD, vì $D \to C$. Dễ thấy có thể tách LĐQH p thành 2 LĐQH thành phần độc lập nhau là p_1=(ABE, {EA→B, A→BE}) với khoá K_1= A và p_2 =(CD, {C→D}) với khoá K_2 = C. Khi thêm PTH $D \to C$ cho LĐQH p_2 ta có thêm khoá K_3 = D. Hợp nhất 2 lược đồ này lại ta sẽ được đúng 2 khoá là

K_1K_2 = AC và K_1K_3 = AD.

BÀI GIẢI

ĐỀ THI TUYỂN CAO HỌC NGÀNH GIS NĂM 2003
ĐẠI HỌC BÁCH KHOA TP HỒ CHÍ MINH
Môn: Cơ sở dữ liệu, đề số 3

Thời gian 120 phút, không tham khảo tài liệu

Trong đề thi, tập các thuộc tính được viết dưới dạng xâu ký tự, ký hiệu XY được hiểu là phép hợp 2 tập X và Y, XY = X∪Y. Các từ viết tắt: LĐQH: lược đồ quan hệ, PTH: phụ thuộc hàm.

Câu 1.

a) Định nghĩa phép toán lấy giao hai quan hệ:

Phép giao (theo lý thuyết tập hợp hoặc *lấy phần chung*) hai quan hệ tương thích $R(U)$ và $S(U)$, ký hiệu $R\&S$, cho ta quan hệ chứa các bộ xuất hiện đồng thời trong cả hai quan hệ thành phần,

$$P(U) = R\&S = \{ t \mid t \in R,\ t \in S \}$$

b) Thuật toán giao hai quan hệ:

```
Algorithm Intersection
Format: P = R&S
Input:      - Quan hệ R(U)
            - Quan hệ S(U)
Output: - Quan hệ R&S={t|t∈R, t∈S}
```

```
Method
    Create(P,Attr(R));
      for each tuple u in R
            with u in S do
                  add u to P;
      endfor;
      return P;
end Intersection;
```

Câu 2.

a) Định nghĩa bao đóng của tập thuộc tính trong LĐQH: Cho LĐQH p = (U,F), trong đó U là tập thuộc tính, F là tập PTH trên U. Bao đóng của tập thuộc tính X, ký hiệu X^+ là tập thuộc tính

$$X^+ = \{A \in U \mid X \rightarrow A \in F^+\}$$

c) Thuật toán tìm bao đóng của tập thuộc tính trong LĐQH:

Ý tưởng: *Cho tập PTH F trên tập thuộc tính U và một tập con các thuộc tính X trong U. Để xác định bao đóng của tập thuộc tính X, X^+ ta xuất phát từ tập X và bổ sung dần cho X các thuộc tính thuộc vế phải R của các PTH $L \rightarrow R \in F$ thỏa điều kiện $L \subseteq X$. Thuật toán sẽ dừng khi không thể bổ sung thêm thuộc tính nào cho X.*

```
Algorithm Closure
Format: C = X+
Input:        - LĐQH p =(U,F)
              - Tập thuộc tính X ⊆ U
```

```
Output: - Y = X+ = {A ∈ U | X → A ∈ F+}
Method
        Y:=X;
    repeat
            Z:=Y;
            for each FD L→R in F do
                if L ⊆ Y then
                    Y:=Y ∪ R;
                endif;
            endfor;
        until Y=Z;
        return Y;
end Closure;
```

Câu 3.

Giả thiết:

LĐQH p = (U, F); U = ABCDE;

F = { AB → C, D → E, B → CA }

a) Tìm một khoá của lược

Ta tìm tập *M* là giao của to ộ các khoá của *p:*

$$M = U - \bigcup_{L \to R \in F} (R - L)$$

M = ABCDE - CAE = BD, vì (=U nên p có một khoá

M = BD.

01.07.20 Xuất XB do ng và nộp

b) Tập DEB có phải là khoá của p không? Vì sao?

Không, vì DEB chứa thực sự khoá M = BD.

c) Tập CA có phải là khoá của p không? Vì sao?

Không vì p chỉ có một khoá duy nhất M = BD.

d) Lược đồ p còn khoá nào nữa không? Vì sao?

Không, với lý do trên.

e) Tính $Z = (X^+Y)^+ \cap K^+$ *biết X = AB, Y = CA, K là một siêu khoá của p.*

Nhận xét: Vì K là khoá của p nên $K^+ = U$. *Mặt khác, theo công thức* $(X^+Y)^+ = (XY)^+$ *nên ta có*

$Z = (XY)^+ \cap U = (XY)^+ = (ABC)^+ = ABC$.

f) Có thể thêm vào F một PTH để thu được lược đồ có đúng 2 khoá không. Giải thích cách làm?

Thêm, chẳng hạn PTH $E \rightarrow D$ *Khi đó ngoài khoá BD lược đồ có thêm khoá BE, vì* $E \rightarrow D$. *Dễ thấy có thể tách LĐQH p thành 2 LĐQH thành phần độc lập nhau là* p_1=(ABC,{AB→C, B→CA}) *với khoá* K_1= *B và* p_2 =(DE, {D→E}) *với khoá* K_2 = *D. Khi thêm PTH* $E \rightarrow D$ *cho LĐQH* p_2 *ta có thêm khoá* K_3 = *E. Hợp nhất 2 lược đồ này lại ta sẽ được đúng 2 khoá là*

K_1K_2 = *BD và* K_1K_3 = *BE.*

01.07.2003

các khoá của p:

U nên p có một khoá

BÀI TẬP CƠ SỞ DỮ LIỆU

Chịu trách nhiệm xuất bản:
CÁT VĂN THÀNH

In 1000 cuốn khổ 14,5 × 20,5 cm tại Xưởng in Nhà xuất bản Thống kê. Giấy phép xuất bản số: 155-205/XB-QLXB do Cục xuất bản cấp ngày 03 tháng 03 năm 2003. In xong và nộp lưu chiểu tháng 10 năm 2003.