Bài 1, Phần 2: Cơ sở dữ liệu suy diễn, Datalog

PGS.TS. Đỗ Phúc
Khoa Hệ thống thông tin
Trường Đại học Công nghệ Thông tin, ĐHQG-HCM

CSDL so với diễn giải

Mô hình quan hệ

- Quan hệ
- Khóa
- Các dạng chuẩn
- Ràng buộc toàn vẹn

Đại số quan hệ : Chọn, chiếu, kết

Ngôn ngữ SQL

a) Truy vấn

- - Câu truy vấn mở: trả lời Y/N
- - Câu truy vấn đóng: trả về tập các bộ

b) Views là quan hệ không được lưu trữ trong CSDL và được tạo qua các biểu thức

- SELECT Name,Age FROM Person WHERE Age >= 10

Mô hình quan hệ dựa trên logic

- Quan hệ được định nghĩa dưới dạng các công thức wff(well formed formulas)

  person(ols,name,age,salary)
- Hàm là trường hợp đặc biệt của quan hệ
- Các thông tin

  + Các vị từ EDB

  person(0111,'Albert',xage,xsalary)

  + Các vị từ IDB

person(x,y,z,45) :- person(X,Y,Z,W) & W >= 35

Ý nghĩa của luật

Ba cách diễn giải luật:

- Diễn giải theo lý thuyết chứng minh
- Diễn giải theo lý thuyết mô hình
- Diễn giải theo tính toán

Diễn giải theo lý thuyết chứng minh

- Các tiên đề
- Thông tin tường minh, ví dụ age(Albert,20)
- Thông tin ẩn được suy từ các vị từ EDB và IDB
- Phép phủ định
- Vị từ khẳng định: ví dụ: age(Albert,30)
- Vị từ phủ định: ví dụ: ~age(Albert,30)

- **b) Chứng minh**
- Tất cả các sự kiện (facts) suy được qua các vị từ IDB là suy được bằng modun ponen
- person(0111,'Albert',44,xsalary)
- person(x,y,z,45) :- person(x,y,z,w) & z >= 35

- person(0111,'Albert',44,45)
- **c) Nghĩa của tập luật** theo lý thuyết chứng minh là tập các sự kiện được suy từ các sự kiện cho trước hay có trong CSDL, dùng các luật theo hướng chuỗi "tiến" , nghĩa là chiều suy diễn từ bên phải sang trái.

Diễn giải luật theo lý thuyết mô hình

- Các luật được định nghĩa theo thế giới khả dĩ hay mô hình
- Diễn giải: gán giá trị chân lý đúng hay sai cho các thể hiện khả dĩ của vị từ
- Mô hình của tập các luật là một diễn giải làm cho luật đúng từ phép gán các trị trong miền trị cho các biến trong từng luật

Ví dụ

Cho p, q và r là các vị từ sau:

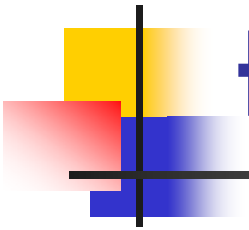
- p(x) :- q(x) (1)
- q(x) :- r(x) (2)

Cơ sở dữ liệu {r1}

- Miền trị nguyên
- Các diễn giải
- M1={r(1),q(1),p(1),q(2),p(2),p(3)}
- M2={r(1),q(1),p(1)}
- M3={r(1),q(2),p(2)}
- Lưu ý : p->q = ~p V q , chỉ sai khi p đúng và q sai

Mô hình cực tiểu

- Gọi M1,..,Mn là các mô hình của tập các công thức wff.
- Mô hình tối thiểu của S là mô hình Mk sao cho:
- $Mk \subseteq Mj$, $j \in \{1,\ldots,k-1,k+1,\ldots,n\}$
- $\sim\exists$M, M là mô hình của S và M $\subseteq$ Mk



Định nghĩa tính toán

- Các thuật toán để thẩm định tính đúng/sai của luật
- Prolog có các thuật toán để tìm chứng minh cho sự kiện

Các khía cạnh của Datalog

- Mô hình dựa trên logic
- Tính toán trên đồ thị phụ thuộc
- Tính toán trên mô hình tối thiểu

Mô hình datalog

- Tiếp cận lý thuyết mô hình
- Các phát biểu Prolog: công thức nguyên tử, ký hiệu vị từ, hàm sinh trị
- Quy ước Prolog: ký hiệu hàm, hằng
- Phát biểu logic (mệnh đề Horn)
- **B :- A1 & A2 & . . . . & An**
- Nếu A1 & A2 & . . . . & An thì B
- A1 & A2 & . . . . & An phần thân (body) của luật
- B là phần đầu (head) của luật

Trong CSDL

- Vị từ EDB: các quan hệ được lưu trong CSDL
- Các vị từ cài sẵn (built –in)
- Các vị từ IDB: mệnh đề Horn suy ra các view

Thẩm định các luật không đệ qui

- Không phủ định
- Đổi sang biểu thức đại số quan hệ

Tính quan hệ được suy

- Đối với từng luật r có pi ở phần đầu, tính quan hệ cho phần thân của luật. Phép tính sử dụng là phép kết tự nhiên theo các đích con khác nhau.
- Tính quan hệ IDB của pi.

Quan hệ được định nghĩa qua phần thân của luật

- Quan hệ r cho luật q :- p1 & . . . & pn.
- Pj ={ (a1, . . . ,ak) / p(a1, . . . ,ak) là đúng }
- Đích con S của luật r được biểu diễn bằng phép thay thế nếu thỏa
- Nếu S là đích con thông thường, thì S trở thành p(b1, . . . ,bk) với (b1, . . . ,bk) là một bộ trong quan hệ P ứng với p.
- Nếu S là đích con cài sẵn thì với phép thế S trở thành, b θ c, quan hệ số học b θ c là đúng

Ví dụ

- cousin(X,Y) :- parent(X,Xp) & parent(Y,Yp) & sibling(Xp,Yp)
- Đã tính xong sibling, parent
- R(X,Xp,Y,Yp) = P(X,Xp) ∞ P(Y,Yp) ∞ S(Xp,Yp)
- Bộ của R có dạng (a,b,c,d)

  với (a,b) thuộc P

  (c,d) thuộc P và

  (b,d) thuộc S

Với luật:
sibling(X,Y) :- parent(X,Z) & parent(Y,Z) & X ≠ Y

- $Q(X,Y,Z) = \sigma_{X \neq Y} (P(X,Z) \infty P(Y,Z))$
- P(X,Y) :- q(a,X) & r(X,Z,X) & s(Y,Z)
- Quan hệ q(a,X): $T(X) = \Pi_{\$1} (\sigma_{\$1 = a}(Q))$
- Quan hệ q(X,Z,X): $U(X,Z) = \Pi_{\$1,\$2} (\sigma_{\$1 = \$3}(R))$
- Quan hệ: s(Y,Z): S(Y,Z)
- Với $k là thuộc tính thứ k trong quan hệ

Thuật toán 1

- **Nhập:** Phần thân của một luật datalog r có chứa các đích con S1,…,Sn và các biến X1,…,Xn. Với mỗi Si = pi(Ai1, . . . , Aik) là vị từ thông thường sẽ có một quan hệ đã được tính R trong đó có A là đối, biến hoặc hằng.
- **Xuất:** Biểu thức đại số quan hệ, ký hiệu là

  EVAl_RULE(r,R1,…,Rn)
- cho phép tính từ các quan hệ R1,…,Rn tính được một quan hệ R(X1,…,Xm) có chứa các bộ (a1,…,am) sao cho khi thay aj vào Xj, 1 <= j <= m , tất cả đích con S1,…,Sn đều đúng.

Phương pháp

- Biểu thức được xây dựng qua các bước sau:
- 1.Đối với mỗi đích con thông thường Si, gọi Qi là biểu thức $\Pi_{Vi}(\sigma_{Fi}(Ri))$. Với Vi là tập hợp các thành phần chỉ chứa đúng một xuất hiện của một biến X có trong đối của Si. Công thức Fi là phép AND của các điều kiện sau:
- Nếu ở vị trí k của Si có hằng a thì Fi có điều kiện $k = $l
- Nếu ở vị trí k và l của Si chứa các giá trị giống nhau thì Fi có điều kiện $k= $l
- Đặc biệt nếu Fi không có một điều kiện nào, chẳng hạn khi Si = p(X,Y) thì xem Fi là điều kiện đồng nhất đúng, như thế Qi = Ri.

- 2.Đối với mỗi biến X không hiện diện trong các đích con thông thường, hãy tính biểu thức Dx nhằm tạo ra quan hệ một ngôi chứa tất cả các giá trị mà X có thể nhận trong phép gán làm thỏa tất cả đích của luật r. Do luật r là an toàn nên X phải được gán bằng với biến Y có giới hạn nào đó qua một chuỗi các phép gán bằng "=" và Y được giới hạn nhờ một hằng a nào đó trong đích con, hoặc Y là một đối của một đích con thông thường.
- Nếu Y = a là một đích con, thì đặt Dx là biểu thức hằng {a}
- Nếu Y xuất hiện như là đối thứ j của một đích con thông thường Si, thì đặt Dx là $\Pi j(Ri)$

- 3. Gọi E là nối tự nhiên của tất cả các Qi được định nghĩa trong (1) và các Dx được định nghĩa trong (2). Trong phép nối này, ta xem Qi là quan hệ với các thuộc tính là các biến trong Si , và xem Dx là quan hệ có thuộc tính X
- 4. Gọi EVAl_RULE(r,R1,…,Rn) là σF(E) trong đó F là hội các biểu thức XθY tương ứng với các đích con cài sẵn (built-in), XθY xuất hiện trong số các đích con p1, … , pm và E là biểu thức được xây dựng trong bước (3). Nếu không có đích con cài sẵn nào thì biểu thức cuối cùng chính là E.

Định lý 1

- Thuật toán 1 là đúng theo nghĩa quan hệ R được tạo ra có tất cả và chỉ những bộ (a1, . . . , am) sao cho khi thay thế mỗi Xj bằng aj, mỗi đích con Si đều được làm đúng.
- (Xem chứng minh trong JD Ullman, Vol1, chương 3)

Ví dụ

P(X,Z) :- q(a,X) & r(X,Z,X) & s(Y,Z)

- Đích con: S1 là q(a,X)
- Q1 = Π$2(σ$1=a (Q))
- Đích con: S2 là r(X,Z,X)
- Q2 = Π$1,$2(σ$1=$2 (R)) = U(X,Z)
- Đích con: S3 là s(X,Z)
- Q3 = S(X,Z)
- Không có vị từ cài sẵn (bước 2 trong thuật toán 1):
- Q1 ∞ Q2 ∞ Q3

Tinh chỉnh luật

- Tính quan hệ cho vị từ nằm ở phần đầu p của luật
- Xét các luật có p trong phần đầu
- Tính quan hệ trong phần thân
- Chiếu các quan hệ xuất hiện trong phần đầu
- Hợp các kết quả
- Vị từ được chỉnh lý của vị từ p là một vị từ p(X1,…,Xk) sao cho:
- Các đối Xj là khác nhau
- Đưa ra các biến mới cho các vị từ trong phần đầu của luật
- Xử lý các đích con cài sẵn trong thân

Cách tạo luật được chỉnh

- Luật r có vị từ trong phần đầu là p(Y1, . . . , Yk) tạo vị từ trong phần đầu với p(X1,…,Xk) với:
- Các biến Xj là khác nhau
- Xi=Xj trong thân với Yi là hằng
- **Ví dụ:** Xét các luật
- p(a,X,Y) :- r(X,Y)
- p(X,Y,X) :- r(Y,X)
- Các luật được chỉnh lý:
- p(U,V,W) :- r(U,W) & U=a
- p(U,V,W) = r(V,U) & W = U
- Tính các quan hệ cho các vị từ không đệ qui

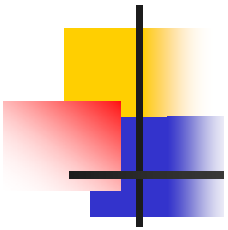
Thuật toán 2

- **Nhập:** Một chương trình Datalog không đệ qui và một quan hệ cho mỗi vị từ EDB hiện diện trong chương trình.
- **Xuất:** Đối với mỗi vị từ IDB p, cho ra một biểu thức đại số quan hệ biểu diễn một quan hệ cho p theo các quan hệ R1, . . . , Rm cho các vị từ EDB.

Phương pháp

- Khởi đầu, chúng ta sẽ tinh chỉnh tất cả các luật. Kế đến chúng ta tạo đồ thị phụ thuộc cho chương trình nhập và sắp thứ tự các vị từ p1, … , pn sao cho nếu đồ thị phụ thuộc của chương trình có một cung từ pi đến pj thì i < j. Ta có thể tìm được một thứ tự như thế vì chương trình nhập là không đệ qui nên đồ thị phụ thuộc là không có chu trình.
- Với i=1,2, … , n chúng ta tạo ra biểu thức của quan hệ Pi(cho pi) như sau:

- Nếu pi là vị từ EDB, gọi Pi là quan hệ cho pi. Ngược lại, giả sử pi là một vị từ IDB thì:
- Đối với từng luật r có pi là phần đầu, hãy dùng thuật toán 1 để tìm biểu thức Er, tính được quan hệ Rr cho thân của luật r theo những quan hệ của các vị từ xuất hiện trong thân của r.



- Do chương trình không đệ qui, tất cả các vị từ xuất hiện trong thân của luật r đều có những biểu thức cho các quan hệ của chúng, được tính thao các quan hệ EDB. Hãy thay các biểu thức thích hợp cho mỗi xuất hiện của một quan hệ IDB trong biểu thức Er để có được một biểu thức mới Fr.
- Đặt lại tên cho các biến nếu cần, chúng ta có thể giả sử phần đầu của một luật cho pi là pi(X1,…,Xk). Sau đó gán biểu thức cho Pi là hợp trên tất cả các luật r cho pi , nghĩa là của các ΠX1, …, Xk (Fr)

Định lý 2

- Thuật toán 2 đúng và cho phép tính chính xác quan hệ cho từng vị từ theo nghĩa là biểu thức do nó xây dựng cho mỗi vị từ IDB sẽ tạo ra:
- Tập các sự kiện (facts) cho vị từ đó mà có thể chứng minh từ CSDL
- Mô hình cực tiểu duy nhất của luật

Ví dụ

Cho quan hệ EDB và IDB như sau:

- p(a,Y) :- s(X,Y)
- p(X,Y) :- s(X,Z) & r(Z,Y)
- q(X,X) :- p(X,b)
- q(X,Y) :- p(X,Z) & s(Z,Y)

Chỉnh lý luật:

- p(X,Y) :- s(X,Y) & X = a
- p(X,Y) :- s(X,Z) & r(Z,Y)
- q(X,Y) :- p(X,b) & X=Y
- q(X,Y) :- p(X,Z) & s(Z,Y)

- Khởi đầu bằng p và q phụ thuộc vào p
- Dùng thuật toán 1, tính quan hệ trong phần thân của luật
- **Đối với vị từ p:**
- P(X,Y) :- ΠX,Y(R(Z,Y) ∞ {a}(X)) $\cup$ $\Pi_{X,Y}$(S(X,Z) ∞ R(Z,Y))
-
- **Đối với vị từ q:**
- q(X,Y) :- p(X,b) & X=Y (1)
- Ta có: $\Pi_{X,Y}(\sigma_{Z=b}(P(X,Z))$ x $\Pi_Y(P(Y,W))$
- q(X,Y) :- p(X,Z) & s(Z,Y) (2)

- Q(X,Y) = $\sigma_{X=Y}$(ΠX ($\sigma_{Z=b}$ (P(X,Z)) x Π_Y(P(Y,W))
- $\cup$ $\Pi_{X,Y}$ ((P(X,Z)) ∞ S(Z,Y))

Datalog đệ qui

- Thuật toán 2 cho các chương trình datalog (sắp thứ tự ví từ)
- Các tình huống pi sẽ được tính trước qj.
- Sườn tổng quát
- Chương trình datalog
- Các quan hệ EDB R1,…,Rk
- Các quan hệ IDB P1,…,Pm
- Pi = EVAL(pi, R1,…,Rk, P1,…,Pm)
- với EVAL = $\cup$pi EVAL_RULE(pi)
- Khởi đầu Pi = $\varnothing$ và do EVAL là "đơn điệu", chúng ta sẽ đi đến một điểm mà không có sự kiện nào có thể bổ sung vào bất kỳ Pi nào

Điểm bất động của các phương trình Datalog

- Phương trình Datalog: thay thế dấu ":-" (if) giữa các vị từ bằng dấu "=" giữa các quan hệ.
- Gọi R1,…,Rk là các quan hệ cho các vị từ EDB, **điểm bất động**(fixed points) của chương trình datalog (ứng với R1,…,Rk) là **lời giải** cho các quan hệ ứng với các vị từ IDB của các phương trình đó.
- Các điểm bất động P1, …, Pm ứng với R1,..,Rk cùng với các luật tạo thành mô hình.
  - Gọi M là mô hình theo đó chỉ có các bộ P1,…,Pm và R1,…,Rk là đúng
  - Bất ký một phép gán làm cho thân của luật r đúng thì đầu của luật r cũng đúng (p(a1,…,an))
  - thế thì (a1,…,an) nằm trong quan hệ cho vị từ IDB p, ngược lại M không phải là điểm bất động.

- Mọi mô hình không phải là điểm bất động vì một mô hình có thể có vài sự kiện xuất hiện trong về trái của phương trình
- Có thể chứng minh mô hình tối thiểu là điểm bất động
- Mô hình tối thiểu duy nhất P0 là điểm bất động duy nhất.

Giải phương trình Datalog đệ qui

- Bắt đầu bằng các quan hệ Pi =∅ cho các vị từ IDB
- Cho sẵn các quan hệ R1,…,Rk cho các vị từ EDB
- **Nguyên tắc:**
- Đầu tiên áp dụng thuật toán 2 cho Pi = ∅ và Ri. Sau đó chèn các bộ mới vào quan hệ IDB
- Lặp lại tiến trình cho đến khi tìm được một bước mà kết quả không thay đổi
- Thuật toán 3:
- Nhập: Một tập các luật datalog với những vị từ EDB r1,…,rk và những vị từ IDB p1,…,pm và một danh sách các quan hệ R1,…,Rk làm giá trị của các vị từ EDB.
- Xuất: Điểm bất động nhỏ nhất (lời giải) của phương trình datalog thu được từ các luật nhập vào

Phương pháp

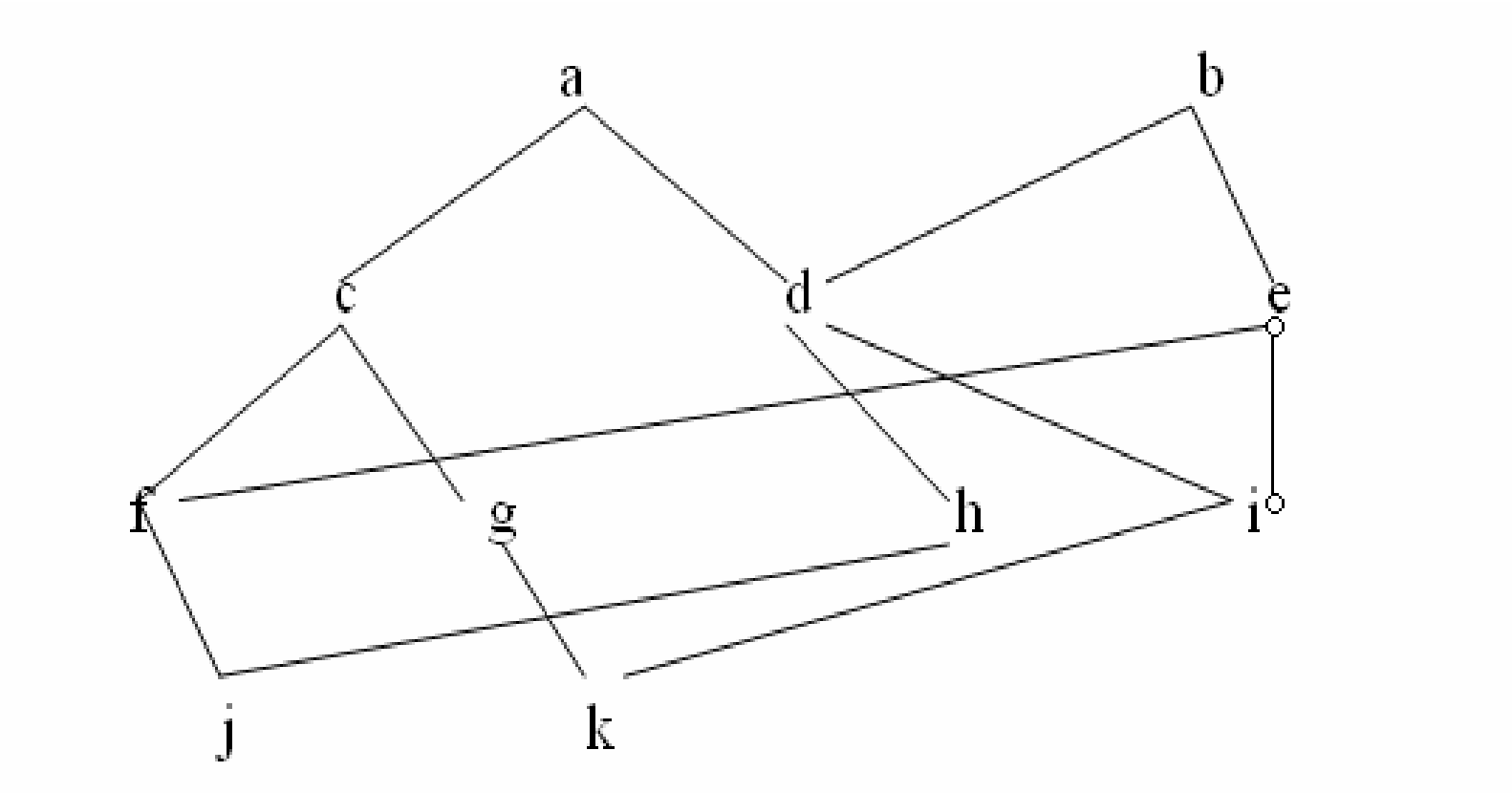
- Trước tiên cần xây dựng các phương trình cho các luật. Những phương trình sẽ có các biến P1,…,Pm tương ứng với các vị từ IDB và phương trình cho Pi là Pi=EVAL(pi,R1…,Rk,P1,…,Pm). Sau đó khởi gán giá trị rỗng cho từng Pi rồi lặp đi lặp lại việc tính EVAL để thu được các giá trị mới cho các Pi.
- Khi không còn bổ sung được nữa, ta sẽ có kết quả mong muốn.

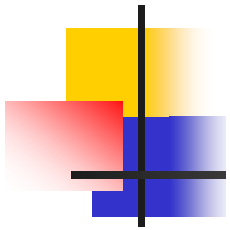
Chi tiết thuật toán

- For i:= 1 to do
- Pi = ∅ ;
- Repeat
- For i:= 1 to m do
- Qi := Pi ; // Lưu giá trị cũ
- For j:= 1 to m do
- Pi := EVAL(Pi,R1,…,Rk,Q1,…,Qm) ;
- Until Pi = Qi với mọi i =1, …, m
- Xuất Pi.

- **Ví dụ:** cho các luật
- sibling(X,Y) :- parent(X,Z) & parent(Y,Z) & X ≠ Y
- cousin(X,Y):- parent(X,Xp) & parent(Y,Yp) & sibling(Xp,Yp)
- cousin(X,Y):- parent(X,Xp) & parent(Y,Yp) & cousin(Xp,Yp)
- related(X,Y) :- sibling(X,Y)
- related(X,Y):- related(X,Z) & parent(Y,Z)
- related(X,Y):- related(Z,Y) & parent(X,Z)

Các bộ của quan hệ Parent(X,Y) được định nghĩa:





- $S(X,Y) = \Pi_{X,Y} (\sigma_{X \neq Y}(P(X,Z) \infty P(Y,Z)))$
- $C(X,Y) = \Pi_{X,Y} (P(X,Xp) \infty P(Y,Yp) \infty S(Xp,Yp)) \cup \Pi_{X,Y} (P(X,Xp) \infty P(Y,Yp) \infty C(Xp,Yp))$
- $R(X,Y) = S(X,Y) \cup \Pi X,Y (R(X,Z) \infty P(Y,Z)) \cup \Pi X,Y (R(Z,Y) \infty P(X,Z))$

Ứng dụng thuật toán ta có các bước và giá trị sau

Bước	S	C	R
1	cd de fg hi fi		
2		fh fi ii gh gi hi jk	cd de fg hi fi
3		jj kk	df dg ch di ci eh ei gj fk hk ij
4			fh dj gh jk gi dk cj ii ck ej ek
5			fj hj gk ik
6			jj kk