

# *Giáo Trình Cơ Sở Dữ Liệu*

## MỤC LỤC

LỜI NÓI ĐẦU	1
chương 1 <b>TỔNG QUAN VỀ CƠ SỞ DỮ LIỆU</b>	
1.1.    MỘT SỐ KHÁI NIỆM CƠ BẢN	2
1.1.1.    Định nghĩa cơ sở dữ liệu	2
1.1.2.    Ưu điểm của cơ sở dữ liệu	2
1.1.3.    Các đối tượng sử dụng CSDL:	2
1.1.3.    Những vấn đề mà CSDL cần phải giải quyết	2
1.1.5.    Hệ quản trị cơ sở dữ liệu	3
1.1.6.    Các ứng dụng của cơ sở dữ liệu	4
1.2.    CÁC MÔ HÌNH CƠ SỞ DỮ LIỆU	5
1.3.    MÔ HÌNH THỰC THỂ KẾT HỢP	5
1.3.1.    Thực thể	6
1.3.2.    Thuộc tính	6
1.3.3.    Loại thực thể	6
1.3.4.    Khóa	6
1.3.5.    Mối kết hợp	8
BÀI TẬP	11
chương 2 <b>MÔ HÌNH CƠ SỞ DỮ LIỆU QUAN HỆ</b>	
2.1.    CÁC KHÁI NIỆM CƠ BẢN	16
2.1.1.    Thuộc Tính(attribute):	16
2.1.2.    Lược đồ quan hệ (Relation schema)	17
2.1.3.    Quan hệ (Relation)	18
2.1.4.    Bộ (Tuple)	18
2.1.5.    Siêu khoá - Khoá chính	18
2.2.    CHUYỂN TỪ MÔ HÌNH THỰC THỂ KẾT HỢP SANG MÔ HÌNH DỮ LIỆU QUAN HỆ	20
2.3.    CÁC PHÉP TOÁN ĐẠI SỐ TRÊN CÁC QUAN HỆ)	
2.3.1.    Phép hợp (Union)	21
2.3.2.    Phép giao (Intersection)	22

2.3.3.Phép trừ (Minus)	22
2.3.4.Tích Descartes (Cartesian Product)	23
2.3.5.Phép chia hai quan hệ	23
2.3.6.Phép chiếu( Projection)	24
2.3.7.Phép chọn (Selection)	25
2.3.8.Phép $\theta$ - kết	25
2.3.9.Phép kết tự nhiên	26
<b>BÀI TẬP</b>	<b>28</b>
chương 3	<b>NGÔN NGỮ TRUY VẤN DỮ LIỆU</b>
3.1.MỞ ĐẦU	29
3.2.TÌM THÔNG TIN TỪ CÁC CỘT CỦA BẢNG - MỆNH ĐỀ SELECT	32
3.3.CHỌN CÁC DÒNG CỦA BẢNG – MỆNH ĐỀ WHERE	34
3.4.THỨ TỰ THỂ HIỆN CÁC BẢN GHI - MỆNH ĐỀ ORDER BY	36
3.5. CÂU LỆNH SQL LỒNG NHAU	37
3.6.GOM NHÓM DỮ LIỆU– MỆNH ĐỀ GROUP BY	38
<b>BÀI TẬP</b>	<b>41</b>
chương 4	<b>RÀNG BUỘC TOÀN VỆN</b>
4.1 RÀNG BUỘC TOÀN VỆN	45
4.1.1 Khái niệm ràng buộc toàn vẹn	45
4.1.2 Các yếu tố của ràng buộc toàn vẹn	46
4.1.2.1.Điều kiện	46
4.1.2.2.Bối cảnh	46
4.1.2.3.Bảng tầm ảnh hưởng	47
4.1.2.4.Hành động	47
4.2. PHÂN LOẠI RÀNG BUỘC TOÀN VỆN	48
4.2.1.Ràng buộc toàn vẹn có bối cảnh là một quan hệ	50
4.2.1.1.Ràng buộc toàn vẹn liên bộ	50
4.2.1.2.ràng buộc toàn vẹn về miền giá trị	51
4.2.1.3.Ràng Buộc Toàn Vẹn Liên Thuộc Tính	51
4.2.2.Ràng buộc toàn vẹn có bối cảnh là nhiều quan hệ	51

4.2.2.1.Ràng Buộc Toàn Vẹn Về Khoá Ngoại:	51
4.2.2.2.Ràng Buộc Toàn Vẹn Liên Thuộc Tính Liên Quan Hệ	52
4.2.2.3.Ràng Buộc Toàn Vẹn Liên Bộ Liên Quan Hệ	52
BÀI TẬP	53
chương 5	<b>LÝ THUYẾT THIẾT KẾ CƠ SỞ DỮ LIỆU</b>
<b>5.1. CÁC VẤN ĐỀ GẶP PHẢI KHI TỔ CHỨC DỮ LIỆU:</b>	55
<b>5.2. PHỤ THUỘC HÀM</b>	56
5.2.1 Định nghĩa phụ thuộc hàm	56
5.2.2 Cách xác định phụ thuộc hàm cho lược đồ quan hệ	57
5.2.3 Một số tính chất của phụ thuộc hàm - <i>hệ luật dẫn armstrong</i> :	58
5.3 BAO ĐÓNG CỦA TẬP PHỤ THUỘC HÀM F VÀ BAO ĐÓNG CỦA TẬP THUỘC TÍNH X	59
5.3.1.Bao đóng của tập phụ thuộc hàm F	59
5.3.2.Bao đóng của tập thuộc tính X	60
5.3.3.Bài toán thành viên	60
5.3.4.Thuật toán tìm bao đóng của một tập thuộc tính (X)	60
5.4. KHOÁ CỦA LƯỢC ĐỒ QUAN HỆ - MỘT SỐ THUẬT TOÁN TÌM KHOÁ	62
5.4.1.Định nghĩa	62
5.4.2.Thuật toán tìm một khoá của một lược đồ quan hệ Q	63
5.4.3.Thuật toán tìm tất cả các khoá của một lược đồ quan hệ	63
5.5. PHỦ TỐI THIỂU	66
5.5.2.Tập phụ thuộc hàm tương đương	66
5.5.1.Phủ tối thiểu	67
5.5.3.Thuật toán tìm phủ tối thiểu của một tập phụ thuộc hàm	68
5.6. DẠNG CHUẨN CỦA LƯỢC ĐỒ QUAN HỆ	69
5.6.1.Một số khái niệm liên quan đến các dạng chuẩn	70
5.6.2.Dạng chuẩn 1	71
5.6.3.Dạng chuẩn 2	71
5.6.4.Dạng chuẩn 3	72
5.6.5.Dạng chuẩn BC	74

BÀI TẬP	75
BÀI TẬP THỰC HÀNH	80
CÁC BỘ ĐỀ THI KIỂM TRA MÔN CSDL	87
TÀI LIỆU THAM KHẢO	110
MỤC LỤC	111

chương 1

## TỔNG QUAN VỀ CƠ SỞ DỮ LIỆU

### 1.1.MỘT SỐ KHÁI NIỆM CƠ BẢN

#### 1.1.1.Định nghĩa Cơ Sở Dữ Liệu (Data Base)

Cơ sở dữ liệu (CSDL) là một hệ thống các thông tin có cấu trúc được lưu trữ trên các thiết bị như băng từ, đĩa từ,... để có thể thoả mãn yêu cầu khai thác đồng thời của nhiều người sử dụng.

CSDL gắn liền với đại số, logic toán và một số lĩnh vực khác.

#### 1.1.2.Ưu điểm của cơ sở dữ liệu

-Giảm sự trùng lặp thông tin xuống mức thấp nhất và do đó bảo đảm được tính nhất quán và toàn vẹn dữ liệu.

-Đảm bảo dữ liệu có thể truy xuất theo nhiều cách khác nhau.

-Khả năng chia sẻ thông tin cho nhiều người sử dụng.

#### 1.1.3.Những vấn đề mà CSDL cần phải giải quyết

*-Tính chủ quyền của dữ liệu*

Tính chủ quyền của dữ liệu được thể hiện ở phương diện an toàn dữ liệu, khả năng biểu diễn các mối liên hệ ngữ nghĩa của dữ liệu và tính chính xác của dữ liệu. Điều này có nghĩa là người khai thác CSDL phải có nhiệm vụ cập nhật các thông tin mới nhất của CSDL.

*-Tính bảo mật và quyền khai thác thông tin của người sử dụng*

Do có nhiều người được phép khai thác dữ liệu một cách đồng thời, nên cần thiết phải có một cơ chế bảo mật và phân quyền hạn khai thác CSDL. Các hệ điều hành nhiều người sử dụng hay hệ điều hành mạng cục bộ đều có cung cấp cơ chế này.

*-Tranh chấp dữ liệu*

Nhiều người được phép truy nhập cùng một lúc vào tài nguyên dữ liệu của CSDL với những mục đích khác nhau, do đó cần thiết phải có một cơ chế ưu tiên khi truy nhập dữ liệu. Cơ chế ưu tiên có thể được thực hiện bằng việc cấp quyền ưu tiên cho từng người khai thác.

*-Đảm bảo an toàn dữ liệu khi có sự cố*

Việc quản lý dữ liệu tập trung có thể làm tăng khả năng mất mát hoặc sai lệch thông tin khi có sự cố như mất điện đột xuất, hay một phần đĩa lưu trữ CSDL bị hư,... một số hệ điều hành mạng có cung cấp dịch vụ sao lưu ảnh đĩa cứng, tự động kiểm tra và khắc phục lỗi khi có sự cố. Tuy nhiên, bên cạnh dịch vụ của hệ điều hành, để đảm bảo CSDL luôn ổn định, một CSDL nhất thiết phải có một cơ chế khôi phục dữ liệu khi có các sự cố bất ngờ xảy ra.

**1.1.4.Các đối tượng sử dụng CSDL**

-Những người sử dụng CSDL không chuyên về lĩnh vực tin học và CSDL.

-Các chuyên viên CSDL biết khai thác CSDL Những người này có thể xây dựng các ứng dụng khác nhau, phục vụ cho các mục đích khác nhau trên CSDL.

-Những người quản trị CSDL, đó là những người hiểu biết về tin học, về các hệ quản trị CSDL và hệ thống máy tính. Họ là người tổ chức CSDL, do đó họ phải nắm rõ các vấn đề kỹ thuật về CSDL để có thể phục hồi CSDL khi có sự cố. Họ là những người cấp quyền hạn khai thác CSDL, do vậy họ có thể giải quyết được các vấn đề tranh chấp dữ liệu nếu có.

**1.1.5. Hệ Quản Trị Cơ Sở Dữ Liệu (Data Base Management System)**

Để giải quyết tốt những vấn đề mà cách tổ chức CSDL đặt ra như đã nói ở trên, cần thiết phải có những phần mềm chuyên dùng để khai thác chúng. Những phần mềm này được gọi là các hệ quản trị CSDL. Các hệ quản trị CSDL có nhiệm vụ hỗ trợ cho các nhà phân tích thiết kế CSDL cũng như những người khai thác CSDL. Hiện nay trên thị trường phần mềm đã có những hệ quản trị CSDL hỗ trợ được nhiều tiện ích như: MS Access, Visual Foxpro, SQL Server Oracle, ...

Mỗi hệ quản trị CSDL đều được cài đặt dựa trên một mô hình dữ liệu cụ thể. Dù là dựa trên mô hình dữ liệu nào, một hệ quản trị CSDL cũng phải hội đủ các yếu tố sau:

***-Ngôn ngữ giao tiếp giữa người sử dụng và CSDL, bao gồm :***

*Ngôn ngữ mô tả dữ liệu:* Để cho phép khai báo cấu trúc của CSDL, khai báo các mối liên hệ của dữ liệu và các quy tắc quản lý áp đặt lên các dữ liệu đó.

*Ngôn ngữ thao tác dữ liệu:* Cho phép người sử dụng có thể cập nhật dữ liệu (thêm/sửa/xoá)

*Ngôn ngữ truy vấn dữ liệu:* Cho phép người khai thác sử dụng để truy vấn các thông tin cần thiết trong CSDL

*Ngôn ngữ quản lý dữ liệu:* Cho phép những người quản trị hệ thống thay đổi cấu trúc của các bảng dữ liệu, khai báo bảo mật thông tin và cấp quyền hạn khai thác CSDL cho người sử dụng,....

***-Từ điển dữ liệu:***

Dùng để mô tả các ánh xạ liên kết, ghi nhận các thành phần cấu trúc của CSDL, các chương trình ứng dụng, mật mã, quyền hạn sử dụng,...

***-Cơ chế giải quyết vấn đề tranh chấp dữ liệu:***

Mỗi hệ quản trị CSDL cũng có thể cài đặt một cơ chế riêng để giải quyết các vấn đề này. Một số biện pháp sau đây thường được sử dụng: thứ nhất: cấp quyền ưu tiên cho từng người sử dụng; thứ hai: Đánh dấu yêu cầu truy xuất dữ liệu, phân chia thời gian, người nào có yêu cầu trước thì có quyền truy xuất dữ liệu trước,...

***-Hệ quản trị CSDL cũng phải có cơ chế sao lưu (backup) và phục hồi (restore) dữ liệu khi có sự cố xảy ra.***

Điều này có thể thực hiện sau một thời gian nhất định hệ quản trị CSDL sẽ tự động tạo ra một bản sao CSDL, cách này hơi tốn kém, nhất là đối với CSDL lớn.

***-Hệ quản trị CSDL phải cung cấp một giao diện thân thiện, dễ sử dụng.***

### **1.1.6.Các Ứng Dụng Của Cơ Sở Dữ Liệu**

Hiện nay, hầu như CSDL gắn liền với mọi ứng dụng của tin học; chẳng hạn như việc quản lý hệ thống thông tin trong các cơ quan nhà nước, việc lưu trữ và xử lý thông tin trong các doanh nghiệp, trong các lĩnh vực nghiên cứu



khoa học, trong công tác giảng dạy, cũng như trong việc tổ chức thông tin đa phương tiện,...

## 1.2.CÁC MÔ HÌNH DỮ LIỆU

Mô hình dữ liệu là sự trừu tượng hoá môi trường thực. Mỗi loại mô hình dữ liệu đặc trưng cho một cách tiếp cận dữ liệu khác nhau của những nhà phân tích thiết kế CSDL. Mỗi loại mô hình dữ liệu đều có những ưu điểm và những mặt hạn chế của nó, nhưng vẫn có những mô hình dữ liệu nổi trội và được nhiều người quan tâm nghiên cứu.

Sau đây chúng ta sẽ điếm qua lịch sử phát triển của các mô hình dữ liệu.

**Vào những năm sáu mươi**, thế hệ đầu tiên của CSDL ra đời dưới dạng mô hình thực thể kết hợp, mô hình mạng và mô hình phân cấp.

**Vào những năm bảy mươi**, thế hệ thứ hai của CSDL ra đời. Đó là mô hình dữ liệu quan hệ do EF. Codd phát minh. Mô hình này có cấu trúc logic chặt chẽ. Đây là mô hình đã và đang được sử dụng rộng khắp trong công tác quản lý trên phạm vi toàn cầu. Việc nghiên cứu mô hình dữ liệu quan hệ nhằm vào lý thuyết chuẩn hoá các quan hệ và là một công cụ quan trọng trong việc phân tích thiết kế các hệ CSDL hiện nay. Mục đích của nghiên cứu này nhằm bỏ đi các phần tử không bình thường của quan hệ khi thực hiện các phép cập nhật, loại bỏ các phần tử dư thừa.

**Sang thập kỷ tám mươi**, mô hình CSDL thứ ba ra đời, đó là mô hình cơ sở dữ liệu hướng đối tượng, mô hình cơ sở dữ liệu phân tán, mô hình cơ sở dữ liệu suy diễn,...

Trong phần tiếp theo sau đây, chúng tôi sẽ trình bày về mô hình dữ liệu tiêu biểu nhất để thiết kế (bước đầu) một ứng dụng tin học đó là mô hình thực thể kết hợp. Trong các chương còn lại của giáo trình này chúng tôi sẽ trình bày về mô hình dữ liệu quan hệ.

## 1.3.MÔ HÌNH THỰC THỂ KẾT HỢP

Hiện nay mô hình dữ liệu quan hệ thường được dùng trong các hệ quản trị CSDL, đây là mô hình dữ liệu ở mức vật lý. Để thành lập được mô hình này, thường là phải dùng mô hình dữ liệu ở mức quan niệm để đặc tả, một trong

những mô hình ở dạng đó là mô hình thực thể kết hợp (sau đó mới dùng một số quy tắc để chuyển hệ thống từ mô hình này về mô hình dữ liệu quan hệ – các quy tắc này sẽ được nói đến trong mục 2.2).

Sau đây là các khái niệm của mô hình thực thể kết hợp.

### 1.3.1. Thực Thể (entity)

Thực thể là một sự vật tồn tại và phân biệt được, chẳng hạn sinh viên Nguyễn Văn Thành, lớp Cao Đẳng Tin Học 2A, môn học Cơ Sở Dữ Liệu, xe máy có biển số đăng ký 52-0549,... là các ví dụ về thực thể.

### 1.3.2. Thuộc tính (attribute)

Các đặc điểm riêng của thực thể gọi là các *thuộc tính*.

Chẳng hạn các thuộc tính của sinh viên Nguyễn Văn Thành là: mã số sinh viên, giới tính, ngày sinh, hộ khẩu thường trú, lớp đang theo học, ...

(Trong giáo trình này, tên thuộc tính được viết bằng chữ in hoa)

### 1.3.3. Loại thực thể (entity type)

Là tập hợp các thực thể có cùng thuộc tính. Mỗi loại thực thể đều phải được đặt tên sao cho có ý nghĩa. Một loại thực thể được biểu diễn bằng một hình chữ nhật.

Ví dụ các sinh viên có mã sinh viên là “02CĐTH019”, “02CĐTH519”, “02TCTH465”,... nhóm lại thành một loại thực thể, được đặt tên là Sinhvien chẳng hạn.

Tương tự trong ứng dụng quản lý điểm của sinh viên (sẽ được trình bày ngay sau đây) ta có các loại thực thể như Monhoc, Lop, Khoa,...

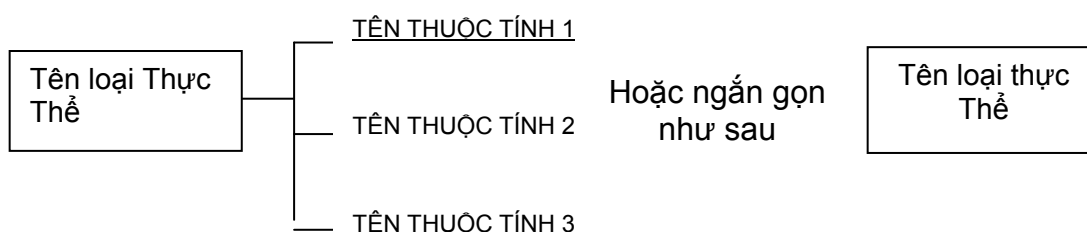
(Trong giáo trình này, tên của loại thực thể được in hoa ký tự đầu tiên, các ký tự còn lại viết thường).

### 1.3.4. Khoá (key)

Khoá của loại thực thể E là một hay một tập các thuộc tính của E có thể dùng để phân biệt hai thực thể bất kỳ của E.

Ví dụ khoá của loại thực thể Sinhvien là MASV, của Lớp là MALOP, của Khoa là MAKHOA, của Monhoc là MAMH,...

Cần chú ý rằng khi biểu diễn một hệ thống bằng mô hình thực thể kết hợp thì tên của các loại thực thể phải khác nhau. Trong danh sách các thuộc tính của một loại thực thể thì tập thuộc tính khoá thường được gạch dưới liền nét. Nếu một hệ thống có nhiều loại thực thể, để đơn giản hoá mô hình, người ta có thể chỉ nêu tên các loại thực thể; còn các thuộc tính của loại thực thể được liệt kê riêng.



### Ví dụ 1.1:

Bài toán quản lý điểm của sinh viên được phát biểu sơ bộ như sau:

Mỗi sinh viên cần quản lý các thông tin như: họ và tên (HOTENSV), ngày tháng năm sinh (NGAYSINH), giới tính (NU), nơi sinh (NƠISINH), hộ khẩu thường trú (TINH). Mỗi sinh viên được cấp một mã số sinh viên duy nhất (MASV) để phân biệt với mọi sinh viên khác của trường, mỗi sinh viên chỉ thuộc về một lớp nào đó.

Mỗi lớp học có một mã số lớp (MALOP) duy nhất để phân biệt với tất cả các lớp học khác trong trường: có một tên gọi (TENLOP) của lớp, mỗi lớp chỉ thuộc về một khoa.

Mỗi khoa có một tên gọi (TENKHOA) và một mã số duy nhất (MAKHOA) để phân biệt với các khoa khác.

Mỗi môn học có một tên gọi (TENMH) cụ thể, được học trong một số đơn vị học trình (DONVIHT) và ứng với môn học là một mã số duy nhất (MAMH) để phân biệt với các môn học khác.

Mỗi giảng viên cần quản lý các thông tin: họ và tên (HOTENGV), cấp học vị (HOCVI), thuộc một chuyên ngành (CHUYENNGANH) và được gán cho một mã số duy nhất gọi là mã giảng viên (MAGV) để phân biệt với các giảng viên

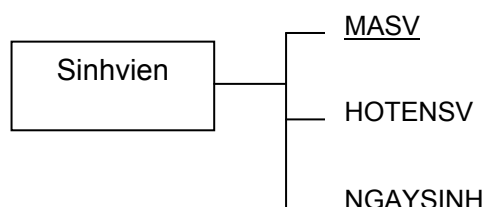
khác. Mỗi giảng viên có thể dạy nhiều môn ở nhiều khoa, nhưng chỉ thuộc về sự quản lý hành chính của một khoa.

Mỗi sinh viên với một môn học được phép thi tối đa 3 lần, mỗi lần thi (LANTHI), điểm thi (DIEMTHI).

Mỗi môn học ở mỗi lớp học chỉ phân công cho một giảng viên dạy (tất nhiên là một giảng viên thì có thể dạy nhiều môn ở một lớp).

Với bài toán trên thì các loại thực thể cần quản lý như: Sinhviên, Môn học, Khoa, Lớp, Giảngviên.

Ví dụ với loại thực thể Sinhviên thì cần quản lý các thuộc tính như: MASV, HOTENSV, NGAYSINH,... và ta có thể biểu diễn như sau:



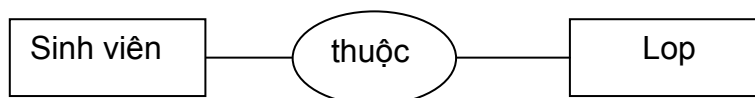
### 1.3.5. Mối Kết Hợp (relationship)

Mối kết hợp diễn tả sự liên hệ giữa các loại thực thể trong một ứng dụng tin học.

Ví dụ mối kết hợp giữa hai loại thực thể Sinhviên và Lop, mối kết hợp giữa Sinhviên với Môn học,...

Mối kết hợp được biểu diễn bằng một hình elip và hai bên là hai nhánh gắn kết với các loại thực thể (hoặc mối kết hợp) liên quan, tên mối kết hợp thường là: *thuộc*, *gồm*, *chứa*,...

Chẳng hạn giữa hai loại thực thể Lớp và Khoa có mối kết hợp “thuộc” như sau:

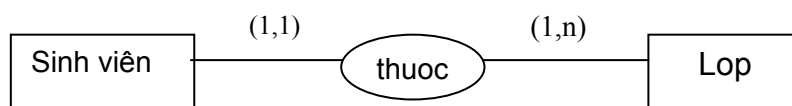


### Bản số của mối kết hợp:

Bản số của một nhánh R trong mối kết hợp thể hiện số lượng các thực thể thuộc thực thể ở nhánh “bên kia” có liên hệ với một thực thể của nhánh R.

Mỗi bản số là một cặp số (min,max), chỉ số lượng tối thiểu và số lượng tối đa của thực thể khi tham gia vào mỗi kết hợp đó.

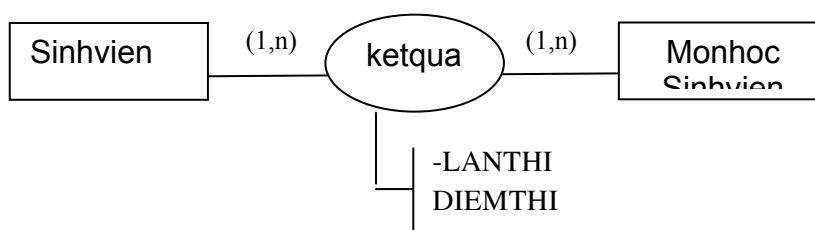
Ví dụ:



Có nghĩa là: “mỗi sinh viên thuộc một và chỉ một lớp nên bản số bên nhánh Sinhviên là (1,1), mỗi lớp có 1 đến n sinh viên nên bản số bên nhánh Lop là (1,n)”

Trong một số trường hợp đặc biệt, mỗi kết hợp có thể có các thuộc tính đi kèm và do đó chúng thường được đặt tên ý với nghĩa đầy đủ hơn.

Ví dụ giữa hai loại thực thể Monhoc và Sinhvien có mỗi kết hợp ketqua với ý nghĩa: “mỗi sinh viên ứng với mỗi lần thi của mỗi môn học có một kết quả điểm thi duy nhất”.



Khoá của mỗi kết hợp: là hợp của các khoá của các loại thực thể liên quan.

Chẳng hạn như thuộc tính MAGV là khoá của loại thực thể Giangvien, MALOP là thuộc tính khoá của loại thực thể Lop, MAMH là thuộc tính khoá của loại thực thể Monhoc, do đó mỗi kết hợp phancong (giữa các loại thực thể Giangvien,Lop,Monhoc) có khoá là {MAGV,MAMH,MALOP} - phancong là mỗi kết hợp 3 ngôi.

(Trong giáo trình này, tên của mỗi kết hợp được viết toàn bằng chữ thường).

Việc thành lập mô hình thực thể kết hợp cho một ứng dụng tin học có thể tiến hành theo các bước sau:

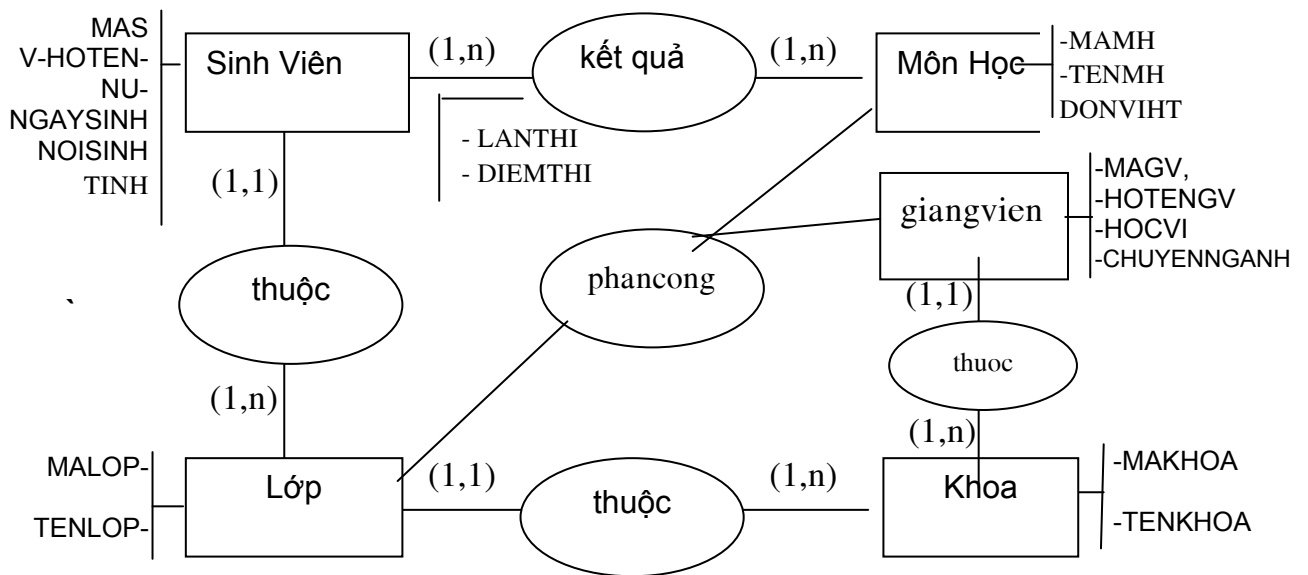
b1.Xác định danh sách các loại thực thể

b2.Xác định các mối kết hợp giữa các loại thực thể để phác thảo mô hình.

b3.Lập bản số của các mối kết hợp.

Để kết thúc chương này, chúng tôi sẽ lập mô hình thực thể kết hợp cho bài toán quản lý điểm của sinh viên đã được nêu trong ví dụ 1.1

**Ví dụ 1.2:**



## BÀI TẬP

Dựa vào các phân tích sơ bộ dưới đây, hãy lập mô hình thực thể kết hợp (gồm loại thực thể, mối kết hợp, bản số, thuộc tính của loại thực thể, khoá của loại thực thể ) cho mỗi bài toán quản lý sau:

### 1.1. QUẢN LÝ SỐ LƯỢNG NGÀY CÔNG CỦA CÁC NHÂN VIÊN

Để quản lý việc phân công các nhân viên tham gia vào xây dựng các công trình. Công ty xây dựng ABC tổ chức quản lý như sau:

Cùng lúc công ty có thể tham gia xây dựng nhiều công trình, mỗi công trình có một mã số công trình duy nhất (MACT), mỗi mã số công trình xác định các thông tin như: Tên gọi công trình (TENCT), địa điểm(ĐIADIEM), ngày công trình được cấp giấy phép xây dựng (NGAYCAPGP), ngày khởi công (NGAYKC), ngày hoàn thành (NGAYHT)

Mỗi nhân viên của công ty ABC có một mã số nhân viên(MANV) duy nhất, một mã số nhân viên xác định các thông tin như: Họ tên (HOTEN), ngày sinh(NGSINH), phái (PHAI), địa chỉ (ĐIACHI),phòng ban, ...

Công ty phân công các nhân viên tham gia vào các công trình, mỗi công trình có thể được phân cho nhiều nhân viên và mỗi nhân viên cùng lúc cũng có thể tham gia vào nhiều công trình. Với mỗi công trình một nhân viên có một số lượng ngày công (SLNGAYCONG) đã tham gia vào công trình đó.

Công ty có nhiều phòng ban(Phòng kế toán, phòng kinh doanh, phòng kỹ thuật, phòng tổ chức, phòng chuyên môn, Phòng phục vụ,...). Mỗi phòng ban có một mã số phòng ban(MAPB) duy nhất, một phòng ban ứng với một tên phòng ban(TENPB).

### 1.2. QUẢN LÝ VIỆC MƯỢN/TRẢ SÁCH Ở MỘT THƯ VIỆN

Một thư viện tổ chức việc cho mượn sách như sau:

Mỗi quyển sách được đánh một mã sách (MASH) dùng để phân biệt với các quyển sách khác (giả sử nếu một tác phẩm có nhiều bản giống nhau hoặc có nhiều tập thì cũng xem là có mã sách khác nhau), mỗi mã sách xác định các thông tin khác như : tên sách (TENSACH), tên tác giả (TACGIA), nhà xuất bản (NHAXB), năm xuất bản (NAMXB).

Mỗi độc giả được thư viện cấp cho một thẻ thư viện, trong đó có ghi rõ mã độc giả (MAĐG), cùng với các thông tin khác như : họ tên (HOTEN), ngày sinh (NGAYSINH), địa chỉ (ĐIACHI), nghề nghiệp(NGHENGHIEP).

Cứ mỗi lượt mượn sách, độc giả phải đăng ký các quyển sách cần mượn vào một phiếu mượn, mỗi phiếu mượn có một số phiếu mượn (SOPM) khác nhau, mỗi phiếu mượn xác định các thông tin như: ngày mượn sách (NGAYMUON), mã độc giả. Các các quyển sách trong cùng một phiếu mượn không nhất thiết phải trả trong một lần. Mỗi quyển sách có thể thuộc nhiều phiếu mượn khác nhau (tất nhiên là tại các thời điểm khác nhau).

### 1.3. QUẢN LÝ LỊCH DẠY CỦA GIÁO VIÊN

Để quản lý lịch dạy của các giáo viên và lịch học của các lớp, một trường tổ chức như sau:

Mỗi giáo viên có một mã số giáo viên (MAGV) duy nhất, mỗi MAGV xác định các thông tin như: họ và tên giáo viên (HOTEN), số điện thoại (DTGV). Mỗi giáo viên có thể dạy nhiều môn cho nhiều khoa nhưng chỉ thuộc sự quản lý hành chính của một khoa nào đó.

Mỗi môn học có một mã số môn học (MAMH) duy nhất, mỗi môn học xác định tên môn học(TENMH). Ứng với mỗi lớp thì mỗi môn học chỉ được phân cho một giáo viên.

Mỗi phòng học có một số phòng học (PHONG) duy nhất, mỗi phòng có một chức năng (CHUCNANG); chẳng hạn như phòng lý thuyết, phòng thực hành máy tính, phòng nghe nhìn, xưởng thực tập cơ khí,...

Mỗi khoa có một mã khoa (MAKHOA) duy nhất, mỗi khoa xác định các thông tin như: tên khoa (TENKHOA), điện thoại khoa(DTKHOA).

Mỗi lớp có một mã lớp (MALOP) duy nhất, mỗi lớp có một tên lớp (TENLOP), sĩ số lớp (SISO). Mỗi lớp có thể học nhiều môn của nhiều khoa nhưng chỉ thuộc sự quản lý hành chính của một khoa nào đó.

Hàng tuần, mỗi giáo viên phải lập lịch báo giảng cho biết giáo viên đó sẽ dạy những lớp nào, ngày nào (NGAYDAY), môn gì?, tại phòng nào, từ tiết nào (TUTIET) đến tiết nào (ĐENTIET), tựa đề bài dạy (BAIDAY), những ghi chú (GHICHU) về các tiết dạy này, đây là giờ dạy lý thuyết (LYTHUYET) hay thực hành - giả sử nếu LYTHUYET=1 thì đó là giờ dạy thực hành và nếu LYTHUYET=2 thì đó là giờ lý thuyết, một ngày có 16 tiết, sáng từ tiết 1 đến tiết 6, chiều từ tiết 7 đến tiết 12, tối từ tiết 13 đến 16.

Một số yêu cầu của hệ thống này như:: Lập lịch dạy trong tuần của các giáo viên. Tổng số dạy của các giáo viên theo từng môn cho từng lớp, ....



#### 1.4. QUẢN LÝ HỌC VIÊN Ở MỘT TRUNG TÂM TIN HỌC

Trung tâm tin học KTCT thường xuyên mở các lớp tin học ngắn hạn và dài hạn. Mỗi lớp ngắn hạn có một hoặc nhiều môn học (chẳng hạn như lớp Tin học văn phòng thì có các môn : Word, Power Point, Excel, còn lớp lập trình Pascal thì chỉ học một môn Pascal). Các lớp dài hạn (chẳng hạn như lớp kỹ thuật viên đồ hoạ đa truyền thông, lớp kỹ thuật viên lập trình, lớp kỹ thuật viên phần cứng và mạng,... ) thì có thể học nhiều học phần và mỗi học phần có thể có nhiều môn học.

Mỗi học viên có một mã học viên(MAHV) duy nhất và chỉ thuộc về một lớp duy nhất (nếu học viên cùng lúc học nhiều lớp thì ứng với mỗi lớp, học viên đó có một MAHV khác nhau). Mỗi học viên xác định họ tên (HOTEN), ngày sinh (NGAYSINH),nơi sinh (NOISINH), phái nam hay nữ (PHAI), nghề nghiệp (NGHENGHIEP) - nghề nghiệp là SINH VIÊN, GIÁO VIÊN, KỸ SƯ, HỌC SINH, BUỒN BÁN,...

Trung tâm KTCT có nhiều lớp, mỗi lớp có một mã lớp duy nhất (MALOP), mỗi lớp xác định các thông tin: tên lớp (TENLOP), thời khoá biểu, ngày khai giảng (NGAYKG), học phí (HOCPHI).

Chú ý rằng tại một thời điểm, trung tâm có thể mở nhiều lớp cho cùng một chương trình học. Với các lớp dài hạn thì ngày khai giảng được xem là ngày bắt đầu của mỗi học phần và HỌC PHÍ là học phí của mỗi học phần, với lớp ngắn hạn thì HỌC PHÍ là học phí của toàn khoá học đó.

Trung tâm có nhiều môn học, mỗi môn học có mã môn học (MAMH) duy nhất, mỗi môn học xác định tên môn học(TENMH), số tiết lý thuyết (SOTIETLT), số tiết thực hành (SOTIETTH).

Mỗi học viên ứng với mỗi môn học có một điểm thi(DIEMTHI) duy nhất. Mỗi lần đóng học phí, học viên sẽ được trung tâm giao cho một phiếu biên lai thu tiền, mỗi biên lai có một số biên lai duy nhất để quản lý.

Một số yêu cầu của hệ thống này như::Lập danh sách những học viên khai giảng khoá ngày nào đó. Lập danh sách các học viên của một lớp ? Cho biết số lượng học viên của mỗi lớp khai giảng khoá ngày nào đó ?

#### 1.5. QUẢN LÝ COI THI TUYỂN SINH

Một hội đồng coi thi tuyển sinh có nhiều điểm thi, mỗi điểm thi được đặt tại một trường nào đó. Các điểm thi (DIEMTHISO) được đánh số là điểm thi số 1, điểm thi số 2, điểm thi số 3,... Mỗi điểm thi xác định địa chỉ (DIACHIDIEMTHI). Ví dụ: điểm thi số 1, đặt tại trường PTTH Nguyễn Thị Minh Khai, điểm thi số 2 đặt tại trường PTTH Bùi Thị Xuân,...

Mỗi thí sinh có một số báo danh (SOBD) duy nhất, mỗi số báo danh xác định các thông tin: họ và tên (HOTEN), ngày sinh (NGAYSINH), phái (PHAI), hộ khẩu thường trú (TINH), đối tượng dự thi (DOITUONG), ngành đăng ký thi, khu vực của thí sinh (KHUVUC), số hiệu phòng thi. Ví dụ: thí sinh Vũ Mạnh Cường, có số báo danh là 02978, sinh ngày 12/12/1984, phái nam, hộ khẩu thường trú tại Chợ Gạo - Tiền Giang, thuộc khu vực 1, đối tượng là 5B, đăng ký dự thi vào ngành có mã ngành là 01, thi tại phòng thi 0178, điểm thi số 1.

Mỗi ngành có một mã ngành (MANGANH) duy nhất, mỗi mã ngành xác định tên ngành (TENNGANH)

Mỗi điểm thi có nhiều phòng thi – mỗi phòng thi (PHONGTHI) được đánh số khác nhau ở tất cả các điểm thi. Trong một phòng thi, danh sách các thí sinh được sắp xếp theo thứ tự alphabet (do đó trong một phòng thi có thể có thí sinh của nhiều ngành khác nhau). Mỗi phòng thi có thêm cột ghi chú (GHICHU) - ghi thêm các thông tin cần thiết như phòng thi đó nằm tại dãy nhà nào. Ví dụ phòng thi 0060 nằm ở dãy nhà H lầu 2 - điểm thi số 1 - trường PTTH Bùi Thị Xuân.

Mỗi môn thi có một mã môn thi duy nhất (MAMT), mỗi mã môn thi biết các thông tin như : tên môn thi (TENMT), ngày thi (NGAYTHI), buổi thi (BUOITHI), thời gian làm bài thi được tính bằng phút (PHUT). Thời gian làm bài thi của các môn tối thiểu là 90 phút và tối đa là 180 phút (tùy theo kỳ tuyển sinh công nhân, trung cấp, cao đẳng hay đại học)

Mỗi ngành có một mã ngành, chẳng hạn ngành Công Nghệ Thông Tin có mã ngành là 01, ngành Công Nghệ Hoá Thực Phẩm có mã ngành là 10,...

Mỗi đơn vị có cán bộ tham gia vào kỳ thi có một mã đơn vị duy nhất (MADONVI), mã đơn vị xác định tên đơn vị (TENDONVI). Nếu là cán bộ, công nhân viên của trường thì đơn vị là khoa/phòng quản lý cán bộ đó, nếu là giáo viên từ các trường khác thì ghi rõ tên đơn vị đó. Chẳng hạn cán bộ Nguyễn Thanh Liêm đơn vị Khoa Công Nghệ Thông Tin, cán bộ coi thi Nguyễn Thị Tuyết Mai, đơn vị trường PTTH Ngôi Sao - Quận 1,...

Mỗi cán bộ coi thi chỉ làm việc tại một điểm thi nào đó. Mỗi cán bộ có một mã số duy nhất (MACANBO), mỗi MACANBO xác định các thông tin khác như : họ và tên (HOTENCB),

đơn vị công tác, chức vụ (CHUCVU) được phân công tại điểm thi, chẳng hạn chức vụ là điểm trưởng, điểm phó, giám sát, thư ký, cán bộ coi thi, phục vụ,... Ví dụ cán bộ Nguyen Van Thanh đơn vị Khoa Công Nghệ Thông Tin, làm nhiệm vụ thi tại điểm thi số 1, chức vụ là giám sát phòng thi.

## chương 2

# MÔ HÌNH DỮ LIỆU QUAN HỆ

### 2.1. CÁC KHÁI NIỆM CƠ BẢN

Mô hình dữ liệu quan hệ (Relational Data Model)- gọi tắt là mô hình quan hệ, do EF.Codd đề xuất năm 1970. Nền tảng lý thuyết của nó là khái niệm lý thuyết tập hợp trên các quan hệ, tức là tập của các bộ giá trị.

Mô hình dữ liệu quan hệ là mô hình được nghiên cứu nhiều nhất, và thực tiễn đã cho thấy rằng nó có cơ sở lý thuyết vững chắc nhất. Mô hình dữ liệu này cùng với mô hình thức thể kết hợp đang được sử dụng rộng rãi trong việc phân tích và thiết kế CSDL hiện nay.

Sau đây là các khái niệm của mô hình dữ liệu quan hệ.

#### 2.1.1.Thuộc Tính(attribute):

Thuộc tính là các đặc điểm riêng của một đối tượng (*đối tượng* được hiểu như là một loại thực thể ở mô hình thực thể kết hợp), mỗi thuộc tính có một *tên gọi* và phải thuộc về một *kiểu dữ liệu* nhất định.

#### **Kiểu dữ liệu** (data type)

Các thuộc tính được phân biệt qua tên gọi và phải thuộc một kiểu dữ liệu nhất định (số, chuỗi, ngày tháng, logic, hình ảnh,...). Kiểu dữ liệu ở đây có thể là kiểu vô hướng hoặc là kiểu có cấu trúc. Nếu thuộc tính có kiểu dữ liệu là vô hướng thì nó được gọi là thuộc tính đơn hay thuộc tính nguyên tố, nếu thuộc tính có kiểu dữ liệu có cấu trúc thì ta nói rằng nó không phải là thuộc tính nguyên tố

Chẳng hạn với sinh viên Nguyễn Văn Thành thì các thuộc tính họ và tên, mã số sinh viên thuộc kiểu chuỗi, thuộc tính ngày sinh thuộc kiểu ngày tháng, hộ khẩu thường trú kiểu chuỗi, thuộc tính hình ảnh kiểu hình ảnh,...

#### **Miền giá trị** (domain of values)

Thông thường mỗi thuộc tính chỉ chọn lấy giá trị trong một tập con của kiểu dữ liệu và tập hợp con đó gọi là miền giá trị của thuộc tính đó. Chẳng hạn thuộc tính NỮ có miền giá trị là {nam,nữ}, thuộc tính màu da có miền giá trị là

{da trắng, da vàng, da đen, da đỏ}, thuộc tính điểm thi là các số thuộc tập  $\{0; 1; 2; \dots, 10\}$ .

Lưu ý rằng nếu không lưu ý đến ngữ nghĩa thì tên của các thuộc tính thường được ký hiệu bằng các chữ cái in hoa đầu tiên trong bảng chữ cái la tinh: A,B,C,D,... Những chữ cái in hoa X,Y,Z,W,... thường dùng thay cho một nhóm nhiều thuộc tính. Đôi khi còn dùng các ký hiệu chữ cái với các chỉ số  $A_1, A_2, \dots, A_n$  để chỉ các thuộc tính trong trường hợp tổng quát hay muốn đề cập đến số lượng các thuộc tính. Tên thuộc tính phải được đặt một cách gọi nhớ, không nên đặt tên thuộc tính quá dài (vì như thế sẽ làm cho việc viết các câu lệnh truy vấn trở nên vất vả hơn), nhưng cũng không nên đặt tên thuộc tính quá ngắn (vì nó sẽ không cho thấy ngữ nghĩa của thuộc tính), đặc biệt không đặt trùng tên hai thuộc tính mang ngữ nghĩa khác nhau thuộc hai đối tượng khác nhau.

Trong nhiều hệ quản trị cơ sở dữ liệu, người ta thường đưa thêm vào miền giá trị của các thuộc tính một giá trị đặc biệt gọi là giá trị rỗng (NULL). Tùy theo ngữ cảnh mà giá trị này có thể đặc trưng cho một giá trị không thể xác định được hoặc một giá trị chưa được xác định ở vào thời điểm nhập tin nhưng có thể được xác định vào một thời điểm khác.

### 2.1.2 Lược Đồ Quan Hệ (relation schema)

Tập tất cả các thuộc tính cần quản lý của một đối tượng cùng với các mối liên hệ giữa chúng được gọi là *lược đồ quan hệ*. Lược đồ quan hệ Q với tập thuộc tính  $\{A_1, A_2, \dots, A_n\}$  được viết là  $Q(A_1, A_2, \dots, A_n)$ , ký hiệu  $Q^+ = \{A_1, A_2, \dots, A_n\}$ .

Chẳng hạn lược đồ quan hệ Sinhviên với các thuộc tính như đã được liệt kê trong ví dụ 1.1 được viết như sau:

Sinhvien(MASV, HOTENSV, NU, NGAYSINH, NOISINH, TINH, MALOP)

Thường thì khi thành lập một lược đồ quan hệ, người thiết kế gán cho nó một ý nghĩa nhất định, gọi là *tân từ của lược đồ quan hệ*. chẳng hạn tân từ của lược đồ quan hệ Sinhvien là: "Mỗi sinh viên có mỗi MASV duy nhất. Mỗi

MASV xác định các thuộc tính còn lại của sinh viên đó như HOTENSV, NU, NGAYSINH, NOISINH, TINH, MALOP”

Khi phát biểu tân từ cho một lược đồ quan hệ, người thiết kế cần phải mô tả đầy đủ ý nghĩa để người khác tránh hiểu nhầm. Dựa vào tân từ này, người ta xác định được tập khoá, siêu khoá của lược đồ quan hệ (sẽ được trình bày trong những mục kế tiếp).

Nhiều lược đồ quan hệ cùng nằm trong một hệ thống thông tin được gọi là một *lược đồ cơ sở dữ liệu*.

Khái niệm lược đồ quan hệ ứng với khái niệm loại thực thể ở mô hình thực thể kết hợp.

### 2.1.3. Quan Hệ (relation)

Sự thể hiện của lược đồ quan hệ ở một thời điểm nào đó được gọi là *quan hệ*, rõ ràng là trên một lược đồ quan hệ có thể xác định nhiều quan hệ. Thường ta dùng các ký hiệu như R, S, Q để chỉ các lược đồ quan hệ, còn quan hệ thường được dùng bởi các ký hiệu là r, s, q, ...

Về trực quan thì quan hệ (hay bảng quan hệ) như là một bảng hai chiều gồm các dòng và các cột.

Một quan hệ có n thuộc tính được gọi là quan hệ n ngôi.

Để chỉ quan hệ r xác định trên lược đồ quan hệ Q ta có thể viết  $r(Q)$ .

### 2.1.4 Bộ (Tuple)

Mỗi *bộ* là những thông tin về một đối tượng thuộc một quan hệ, bộ cũng còn được gọi là mẫu tin.

Thường người ta dùng các chữ cái thường (như t,  $\mu$ , ...) để biểu diễn bộ trong quan hệ, chẳng hạn để nói t là một bộ của quan hệ r thì ta viết  $t \in r$ .

### 2.1.5. Siêu Khoá – Khoá (super key- key)

S là siêu khoá (super key) của Q nếu với r là quan hệ bất kỳ trên Q,  $t_1, t_2$  là hai bộ bất kỳ thuộc r thì  $t_1.S \neq t_2.S$ .

Một lược đồ quan hệ có thể có một hoặc nhiều siêu khoá.

Chẳng hạn lược đồ quan hệ Sinhvien ở trên có các siêu khoá là: {MASV, HOTENSV}, {MASV, HOTENSV, NU}, {MASV, HOTENSV, NU, TINH }, ...

Siêu khoá không chứa một siêu khoá nào khác được gọi là khoá *chỉ định*, trong trường hợp lược đồ quan hệ có nhiều khoá chỉ định (hay khoá nội), thì khoá được chọn để cài đặt gọi là *khoá chính* (trong các phần sau khoá chính được gọi tắt là khoá). Chẳng hạn với lược đồ quan hệ Sinhvien trên có khoá là {MASV}. Thường các thuộc tính khoá được gạch dưới theo kiểu liền nét.

Một thuộc tính được gọi là thuộc tính khoá ngoại nếu nó không là thuộc tính khoá của một lược đồ quan hệ này nhưng lại là thuộc tính khoá của một lược đồ quan hệ khác, chẳng hạn như MALOP là khoá ngoại của lược đồ quan hệ Sinhvien. Thường các thuộc tính khoá ngoại được gạch dưới theo kiểu không liền nét.

Sinhvien(MASV, HOTENSV, NU, NGAYSINH, TINH, MALOP)

Lop(MALOP, TENLOP, MAKHOA)

Ý nghĩa thực tế của khoá là dùng để nhận diện một bộ trong một quan hệ, nghĩa là, khi cần tìm một bộ t nào đó, ta chỉ cần biết giá trị của thành phần khoá của t là đủ để dò tìm và hoàn toàn xác định được nó trong quan hệ.

Trong thực tế đối với các loại thực thể tồn tại khách quan (ví dụ: Sinh viên, Giảng viên, Nhân viên, Hàng hoá,...) người thiết kế cơ sở dữ liệu thường gán thêm cho các lược đồ quan hệ này một thuộc tính giả gọi là mã số để làm khoá (ví dụ: mã số sinh viên, mã số giảng viên, mã số nhân viên, mã số hàng hoá,...). Trong khi đó các lược đồ quan hệ biểu diễn cho sự trừu tượng hoá thường có khoá là một tổ hợp của hai hay nhiều thuộc tính của nó.

Một số hệ quản trị cơ sở dữ liệu hiện nay có tự động kiểm tra tính duy nhất trên khoá chính. Tức là nếu thêm một bộ mới q2 có giá trị khoá chính trùng với giá trị khoá chính của một bộ q1 nào đó đã có trong quan hệ thì hệ thống sẽ báo lỗi và yêu cầu nhập lại một giá trị khác.

Người ta cũng quy ước rằng:

-Trong một bộ của quan hệ các thuộc tính khoá không chứa giá trị rỗng.

-Không được phép sửa đổi giá trị thuộc tính khoá của một bộ q. Nếu muốn sửa đổi giá trị thuộc tính khoá của một bộ q, người sử dụng phải huỷ bỏ bộ q và sau đó thêm một bộ q' với giá trị khoá đã được sửa đổi.

## 2.2.CHUYỂN MÔ HÌNH THỰC THỂ KẾT HỢP SANG MÔ HÌNH DỮ LIỆU QUAN HỆ

Sau đây là một số quy tắc được sử dụng trong việc chuyển đổi mô hình thực thể kết hợp sang mô hình dữ liệu quan hệ.

### Quy tắc 1:

Chuyển đổi mỗi loại thực thể thành một lược đồ quan hệ, các thuộc tính của loại thực thể thành các thuộc tính của lược đồ quan hệ, thuộc tính khoá của loại thực thể là thuộc tính khoá của lược đồ quan hệ.

Chẳng hạn loại thực thể Sinhvien ở ví dụ 1.2 khi áp dụng quy tắc 1 thì sẽ được chuyển thành lược đồ quan hệ Sinhvien như sau:

Sinhvien(MASV, HOTENSV, NU, NGAYSINH, TINH,...)

### Quy tắc 2:

Nếu mỗi kết hợp mà cả hai nhánh của nó đều có bản số max là n thì mỗi kết hợp này sẽ được chuyển thành một lược đồ quan hệ K' gồm các thuộc tính của mỗi kết hợp K, cộng thêm các thuộc tính khoá của hai lược đồ quan hệ A, B tương ứng với hai thực thể tham gia vào mỗi kết hợp. Khoá của lược đồ quan hệ K' gồm cả hai khoá của hai lược đồ quan hệ A và B.

Chẳng hạn mỗi kết hợp Phancong giữa ba loại thực thể Giangvien, Monhoc và Lop được chuyển thành lược đồ quan hệ Phancong và có tập khoá là {MAGV,MAMH,MALOP} như sau:

Phancong(MAGV,MAMH,MALOP)

### Quy tắc 3:

Mỗi kết hợp mà một nhánh có bản số là n (nhánh B) và nhánh còn lại có bản số max là 1 (nhánh A) thì loại bỏ mỗi kết hợp này khỏi mô hình thực thể kết hợp và thêm các thuộc tính khoá của lược đồ tương ứng với loại thực thể ở nhánh B vào lược đồ tương ứng với loại thực thể ở nhánh A (khoá của B sẽ thành khoá ngoại của A). Nếu mỗi kết hợp có các thuộc tính thì những thuộc tính này cũng được thêm vào lược đồ quan hệ tương ứng với loại thực thể ở nhánh A.



Chẳng hạn mỗi kết hợp **thuộc** giữa hai loại thực thể Sinhvien và Lop nên lược đồ quan hệ Sinhvien được sửa thành như sau:

Sinhvien(MASV,HOTENSV,NU,NGAYSINH, TINH,MALOP)

#### Quy tắc 4:

Nếu mỗi kết hợp mà cả hai nhánh đều có bản số max là 1 thì áp dụng quy tắc 3 cho một trong hai nhánh tùy chọn.

#### Ví dụ 2.1:

Sau đây là mô hình dữ liệu quan hệ được chuyển từ mô hình thực thể kết hợp ở ví dụ 1.2.

Sinhvien(MASV, HOTENSV, NU, NGAYSINH, NOISINH,TINH, MALOP)

Lop(MALOP,TENLOP,MAKHOA)

Khoa(MAKHOA,TENKHOA)

Monhoc(MAMH,TENMH,DONVIHT)

Giangvien(MAGV,HOTENGV,HOCVI,CHUYENNGANH,MAKHOA)

Ketqua(MASV, MAMH, LANTHI,DIEMTHI)

Phancong(MALOP,MAMH,MAGV)

### 2.3. NGÔN NGỮ ĐẠI SỐ QUAN HỆ

#### 2.3.1. Phép Hợp 2 quan hệ (Union)

Ta nói hai quan hệ  $r_1$  và  $r_2$  là tương thích nếu chúng được định nghĩa trên cùng một lược đồ quan hệ.

Cho hai quan hệ tương thích  $r_1$  và  $r_2$ . Hợp của hai quan hệ  $r_1$  và  $r_2$  ký hiệu là  $r_1 + r_2$  là một quan hệ trên lược đồ quan hệ Q gồm các phần tử thuộc  $r_1$  hoặc thuộc  $r_2$ , tức là:

$$r_1 + r_2 = \{t / t \in r_1 \text{ hoặc } t \in r_2\}$$

#### Ví dụ 2.2

$r_1$

A	B	C	D
a1	b1	c1	d1
a2	b2	c2	d2
a3	b3	c3	d3
a4	b4	c4	d4

$r_2$

A	B	C	D
x1	y1	z1	v1
a2	b2	c2	d2
x3	y3	z3	v3

Khi đó nội dung của quan hệ  $r_1 + r_2$  là:

A	B	C	D
a1	b1	c1	d1
a2	b2	c2	d2
a3	b3	c3	d3
a4	b4	c4	d4
x1	y1	z1	v1
x3	y3	z3	v3

Do thứ tự trước/sau của các bộ trong các quan hệ là không quan trọng nên ta có:

$$\forall r_1, r_2 \text{ thì } r_1 + r_2 = r_2 + r_1$$

$$\forall r \text{ thì } r + r = r$$

Một cách tổng quát có thể lấy hợp của  $n$  quan hệ tương thích: cho  $n$  quan hệ tương thích  $r_1, r_2, \dots, r_n$

Hợp của  $n$  quan hệ  $r_1, r_2, \dots, r_n$  là một quan hệ  $r_1 + r_2 + \dots + r_n$  gồm các phần tử thuộc  $r_1$  hoặc thuộc  $r_2 \dots$  hoặc thuộc  $r_n$

### 2.3.2. Phép Giao 2 quan hệ (Intersection)

Cho lược đồ quan hệ  $Q(A_1, A_2, \dots, A_n)$ .  $r_1$  và  $r_2$  là hai quan hệ tương thích trên  $Q$ .

Giao của hai quan hệ  $r_1$  và  $r_2$  ký hiệu là  $r_1 * r_2$  là một quan hệ trên  $Q$  gồm các phần tử vừa thuộc  $r_1$  vừa thuộc  $r_2$ .

$$\text{Vậy: } r_1 * r_2 = \{ t / t \in r_1 \text{ và } t \in r_2 \}$$

Chẳng hạn với ví dụ 2.2 ở trên thì  $r_1 * r_2$  là:

A	B	C	D
a2	b2	c2	d2

### 2.3.3. Phép Trừ 2 quan hệ (Minus)

Cho hai quan hệ tương thích  $r_1$  và  $r_2$  có tập thuộc tính  $Q(A_1, A_2, \dots, A_n)$ . Hiệu của  $r_1$  cho  $r_2$  ký hiệu là  $r_1 - r_2$  là một quan hệ trên  $Q$  gồm các phần tử chỉ thuộc  $r_1$  mà không thuộc  $r_2$ , nghĩa là  $r_1 - r_2 = \{ t \in r_1 \text{ và } t \notin r_2 \}$

Chẳng hạn với ví dụ 2.2. thì  $r_1 - r_2$  là:

A	B	C	D
a1	b1	c1	d1
a3	b3	c3	d3
a4	b4	c4	d4

### 2.3.4. Tích Descartes của 2 quan hệ (Cartesian Product)

Cho hai lược đồ quan hệ

$$Q_1(A_1, A_2, \dots, A_n)$$

$$Q_2(B_1, B_2, \dots, B_m)$$

Giả sử  $r_1$ ,  $r_2$  là hai quan hệ trên  $Q_1, Q_2$  tương ứng. Tích Descartes (decac) của  $r_1$  và  $r_2$  ký hiệu là  $r_1 \times r_2$  là quan hệ trên lược đồ quan hệ có tập thuộc tính  $Q = Q_1 \cup Q_2$ .

Vậy quan hệ  $r_1 \times r_2$  là quan hệ trên lược đồ:

$$Q = Q_1 \cup Q_2 = \{ A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m \}$$
 với

$$r_1 \times r_2 = \{ (t_1, t_2) : t_1 \in r_1, t_2 \in r_2 \}$$

**Ví dụ 2.3.** cho  $r_1$  và  $r_2$  là

$r_1$			$r_2$		
A	B	C	E	F	H
6	5	4	1	5	9
7	5	5	4	6	8
7	5	5	7	5	3

Thì kết quả  $r_1 \times r_2$  như sau:

A	B	C	E	F	H
6	5	4	1	5	9
6	5	4	4	6	8
6	5	4	7	5	3
7	5	5	1	5	9
7	5	5	4	6	8
7	5	5	7	5	3

### 2.3.5. phép chia 2 quan hệ:

cho 2 lược đồ quan hệ

$$Q_1(A_1, A_2, \dots, A_n)$$

$$Q_2(B_1, B_2, \dots, B_m)$$

$r$  là quan hệ xác định trên  $Q_1$ ;  $s$  là quan hệ xác định trên  $Q_2$  ( $n > m$  và  $s$  khác rỗng), có  $m$  thuộc tính chung (giống nhau về mặt ngữ nghĩa, hoặc các thuộc tính có thể so sánh được) giữa  $r$  và  $s$ . phép chia 2 quan hệ  $r$  và  $s$  ký hiệu  $r \div s$ , là một quan hệ  $q$  có  $n - m$  thuộc tính được định nghĩa như sau:

$$q = r \div s = \{t / \forall u \in s, (t, u) \in r\}$$

**Ví dụ 2.4:**

<b>r</b>			
<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>
a	b	c	d
a	b	e	f
b	c	e	f
c	d	c	d
c	d	e	f
a	b	d	e

<b>s</b>				<b>r ÷ s</b>	
<b>C</b>	<b>D</b>			<b>A</b>	<b>B</b>
c	d			a	b
e	f			c	d

**2.3.6. Phép Chiếu (projection)**

Cho lược đồ quan hệ  $Q(A_1, A_2, \dots, A_n)$ ,  $r$  là quan hệ trên  $Q$  và  $X \subseteq Q^+$ .

Phép chiếu của  $r$  lên tập thuộc tính  $X$ , ký hiệu là  $r[X]$  (hoặc  $r.X$ ) sẽ tạo thành lược đồ quan hệ  $r'$ , trong đó tập thuộc tính của  $r'$  chính là  $X$  và quan hệ  $r'$  được trích từ  $r$  bằng cách chỉ lấy các thuộc tính có trong  $X$ .

Phép chiếu chính là phép rút trích dữ liệu theo cột. Chẳng hạn với  $r_1$  ở ví dụ 2.2 thì khi đó ta có quan hệ con của  $r_1$  chiếu lên  $X = \{A, C\}$  là:

$r_1[X]$

<b>A</b>	<b>C</b>
a1	c1
a2	c2
a3	c3
a4	c4

**2.3.7. Phép Chọn (Selection)**

Cho lược đồ quan hệ  $Q(A_1, A_2, \dots, A_n)$ ,  $r$  là một quan hệ trên lược đồ quan hệ  $Q$ .  $X$  là một tập con của  $Q^+$  và  $E$  là một mệnh đề logic được phát biểu trên tập  $X$ . Phần tử  $t \in r$  thoả mãn điều kiện  $E$  ký hiệu là  $t(E)$ . Phép chọn từ quan hệ  $r$  theo điều kiện  $E$  (ký hiệu là  $r : E$ ) sẽ tạo thành một quan hệ mới ký hiệu là  $r(E)$ , trong đó  $r(E) = \{t: t \in r \text{ và } t(E)\}$

Phép chọn chính là phép rút trích dữ liệu theo dòng. Chẳng hạn với  $r_2$  ở ví dụ 2.3 và điều kiện  $E$  là: " $F \geq 6$ " thì kết quả  $r_2(E)$  hay  $r_2: "F \geq 6"$  có nội dung là

E	F	H
4	6	8

**2.3.8. Phép  $\theta$  - Kết**

Cho hai lược đồ quan hệ  $Q_1$  và  $Q_2$  như sau

$Q_1(A_1, A_2, \dots, A_n)$

$Q_2(B_1, B_2, \dots, B_m)$

$r$  và  $s$  lần lượt là hai quan hệ trên  $Q_1$  và  $Q_2$ .

$A_i$  và  $B_j$  lần lượt là thuộc tính của  $Q_1, Q_2$  sao cho  $MGT(A_i) = MGT(B_j)$ .  $\theta$  là một trong các phép so sánh ( $=, <, >, \leq, \geq, \neq$ ) trên  $MGT(A_i)$ .

$$A_i \theta B_j$$

Phép  $\theta$  kết giữa  $r$  và  $s$  theo điều kiện  $A_i \theta B_j$  ký hiệu là  $r \mid \theta \mid s$  là một quan hệ trên lược đồ quan hệ có tập thuộc tính là  $Q_1 \cup Q_2$  gồm những bộ thuộc tích Descartes của  $r$  và  $s$  sao  $A_i \theta B_j$ .

$$A_i \theta B_j$$

$r \mid \theta \mid s = \{t_{12} / \exists t_1 \in r_1, \exists t_2 \in r_2 \text{ sao cho } t_{12}.Q_1^+ = t_1; t_{12}.Q_2^+ = t_2; t_{12}.A_i \theta t_{12}.B_j\}$

**Ví dụ 2.5** Cho hai quan hệ  $r_1$  và  $r_2$  như sau:

$r_1$

A	B	C
6	5	4
7	5	5
4	2	6

$r_2$

E	F	H
1	5	9
4	6	8
7	5	3

$A_i$  là thuộc tính B,  $B_j$  là thuộc tính F và  $\theta$  là phép so sánh " $\geq$ ". Ta được kết quả là quan hệ sau:

A	B	C	E	F	H
6	5	4	1	5	9
6	5	4	7	5	3
7	5	5	1	5	9
7	5	5	7	5	3

### 2.3.9. Phép Kết Tự Nhiên (natural join)

Nếu  $\theta$  được sử dụng trong phép kết trên là phép so sánh bằng ( $=$ ) thì gọi là phép *kết bằng*. Hơn nữa nếu  $A_i \equiv B_j$  thì phép kết bằng này được gọi là phép *kết tự nhiên*. Phép kết tự nhiên là phép kết thường dùng nhất trong thực tế.

Ngôn ngữ với các phép toán trên gọi là ngôn ngữ đại số quan hệ.

Sau đây là một ví dụ về ngôn ngữ đại số quan hệ.

#### Ví dụ 2.6

Cho lược đồ CSDL dùng để quản lý điểm sinh viên được mô tả như ở ví dụ 2.1. Hãy thực hiện các yêu cầu sau bằng ngôn ngữ đại số quan hệ:

1. Lập danh sách các sinh viên có mã lớp là CDTH2A, danh sách cần MASV, HOTENSV
2. Lập danh sách sinh viên nữ và có mã khoa là "CNTT", danh sách cần MASV, HOTENSV.
3. Lập bảng điểm thi lần 1 của tất cả các môn cho sinh viên lớp CDTH2A, danh sách cần MASV, HOTENSV, TENMH, DIEMTHI.
4. Lập phiếu điểm thi lần 1 các môn cho sinh viên có MASV="00CDTH189". danh sách cần MAMH, TENMH, DONVIHT, DIEMTHI.

Giải:

1. Sinhvien: MALOP="CDTH2A" [MASV, HOTENSV]  
MALOP
2. (Sinhvien|><| Lop: NU and MAKHOA="CNTT")  
[MASV, HOTENSV]

- MASV                      MAMH
3.     (((Sinhvien |><| Ketqua ) |><| Monhoc): MALOP = "CDTH2A" and  
LANTHI=1) [MASV,HOTENSV,TENMH, DIEMTHI]
- MAMH
4.     (Ketqua |><| Monhoc : MASV='00CDTH189' and LANTHI=1)  
[MAMH,TENMH,DONVIHT,DIEMTHI]

**BÀI TẬP**

**2.1.** Hãy lập mô hình dữ liệu quan hệ cho các bài toán quản lý 1.1, 1.2, 1.3,1.4, 1.5. Hãy xác định khoá cho từng lược đồ cho mỗi bài toán trên.

**2.2.** Cho lược đồ cơ sở dữ liệu

Sinhvien(MASV, HOTENSV, NU, NGAYSINH, NOISINH, TINH, MALOP)

Lop(MALOP, TENLOP, MAKHOA)

Khoa(MAKHOA, TENKHOA)

Monhoc(MAMH, TENMH, DONVIHT)

Giangvien(MAGV, HOTENGV, HOCVI, CHUYENNGANH, MAKHOA)

Ketqua(MASV, MAMH, LANTHI, DIEMTHI)

Phancong(MALOP, MAMH, MAGV)

Thực hiện các yêu cầu sau bằng ngôn ngữ đại số quan hệ:

a. Lập danh sách các sinh viên của lớp có MALOP là CDTH2A, danh sách cần các thông tin:

MASV, HOTENSV, NGAYSINH, TINH.

b. Lập danh sách những sinh viên có hộ khẩu thường trú ở tỉnh “LONG AN”, danh sách cần các thông tin: MASV, HOTENSV, NGAYSINH, TENLOP

c. Lập danh sách các giảng viên có cấp học vị là THAC SY của khoa có MAKHOA là “CNTT”, danh sách cần: MAGV, HOTENGV, CHUYENNGANH.

d. Lập bảng điểm thi lần 1 môn học “869” cho tất cả sinh viên thuộc hai lớp có MALOP là “CDTH2A” và “CDTH2B”, danh sách cần: MASV, HOTENSV, DIEMTHI.

e. Lập danh sách các giảng viên đã dạy lớp CDTH2A, danh sách cần các thông tin: MAGV, HOTENGV, TENKHOA, HOCVI, TENMH.

f. Lập danh sách các môn mà lớp CDTH2A đã học, danh sách cần các thông tin:

MAMH, TENMH, DONVIHT, HOTENGV.

g. Lập danh sách những giảng viên đã dạy sinh viên có MASV là “00CDTH189”, danh sách cần MAGV, HOTENGV, HOCVI, CHUYENNGANH, TENKHOA, TENMH

h. Lập danh sách các sinh viên có mã khoa “CNTT” có điểm thi lần 1 môn học “869” lớn hơn hoặc bằng 8, danh sách cần MASV, HOTENSV, DIEMTHI, TENLOP.



chương 3

## NGÔN NGỮ TRUY VẤN DỮ LIỆU

(Structured Query Language)

### 3.1. MỞ ĐẦU

Mỗi hệ quản trị CSDL đều phải có ngôn ngữ giao tiếp giữa người sử dụng với cơ sở dữ liệu. Ngôn ngữ giao tiếp CSDL gồm các loại sau:

*Ngôn ngữ mô tả dữ liệu* (Data Definition Language –DDL): Cho phép khai báo cấu trúc các bảng của CSDL, khai báo các mối liên hệ của dữ liệu (relationship) và các quy tắc áp đặt lên các dữ liệu đó.

*Ngôn ngữ thao tác dữ liệu* (Data Manipulation Language- DML) cho phép người sử dụng có thể thêm (insert), xoá (delete), sửa (update) dữ liệu trong CSDL.

*Ngôn ngữ truy vấn dữ liệu* (hay ngôn ngữ hỏi đáp có cấu trúc(Structured Query Language-SQL)): Cho phép người sử dụng khai thác CSDL để truy vấn các thông tin cần thiết trong CSDL.

*Ngôn ngữ quản lý dữ liệu* (Data Control Language- DCL): Cho phép những người quản trị hệ thống thay đổi cấu trúc của các bảng dữ liệu, khai báo bảo mật thông tin và cấp quyền khai thác CSDL cho người sử dụng.

Những năm 1975-1976, IBM lần đầu tiên đưa ra hệ quản trị CSDL kiểu quan hệ mang tên SYSTEM-R với ngôn ngữ giao tiếp CSDL là SEQUEL (Structured English Query Language). Năm 1976 ngôn ngữ SEQUEL được cải tiến thành SEQUEL-2, khoảng năm 1978-1979 SEQUEL-2 được cải tiến và đổi tên thành ngôn ngữ truy vấn có cấu trúc (Structured Query Language). Cuối năm 1979 hệ quản trị CSDL được cải tiến thành SYSTEM-R\*. Năm 1986 viện tiêu chuẩn quốc gia Mỹ (American National Standards Institute –ANSI) đã công nhận và chuẩn hoá ngôn ngữ SQL và sau đó tổ chức tiêu chuẩn thế giới (International Standards Organization -ISO) cũng đã công nhận ngôn ngữ này. Đó là chuẩn SQL-86. tới nay SQL đã qua 3 lần chuẩn hoá (1989,1992,1996) để

mở rộng các phép toán và tăng cường khả năng bảo mật và tính toàn vẹn dữ liệu.

Trong chương này chúng ta chỉ nghiên cứu về ngôn ngữ SQL.

Ngôn ngữ truy vấn SQL có tập lệnh khá phong phú để thao tác trên cơ sở dữ liệu. Chẳng hạn lệnh create để tạo các bảng quan hệ, lệnh update để cập nhật dữ liệu, lệnh delete để xoá dữ liệu, lệnh insert để thêm dữ liệu,... Trong chương này, chúng tôi chỉ trình bày với bạn đọc câu lệnh quan trọng nhất của SQL đó là câu lệnh hỏi - tìm kiếm dữ liệu SELECT. Kết quả của lệnh select là một quan hệ, quan hệ kết quả này có thể kết xuất ra màn hình, máy in, hoặc là trên các thiết bị lưu trữ thông tin khác. Để đơn giản trong cách trình bày, ta xem quan hệ để thực hiện câu truy vấn là quan hệ nguồn và quan hệ kết quả của truy vấn là quan hệ đích.

Mỗi câu lệnh SQL có thể được viết trên nhiều dòng và kết thúc lệnh bởi dấu chấm phẩy (;), tuy nhiên từ khoá, tên hàm, tên thuộc tính, tên bảng, tên đối tượng thì không được phép viết tách xuống hàng. Trong vận dụng thực tế, từ khoá, tên thuộc tính, tên bảng, tên đối tượng được viết in hoa hay chữ thường là như nhau.

Cú pháp tổng quát của câu lệnh select như sau:

```
select      distinct /*/danh sách thuộc tính/ <biểu thức>,...  
from        <danh sách các quan hệ>  
where       <biểu thức điều kiện>  
group by    <danh sách thuộc tính>  
having      <điều kiện nhóm>  
order by    <danh sách các thuộc tính [desc]>
```

Trong đó:

<biểu thức> (expression) là sự kết hợp một cách hợp lệ giữa các thuộc tính, các toán tử và các hàm. Sau đây sẽ là các toán tử và hàm thông dụng nhất. (cần chú ý rằng cách sử dụng các toán tử và các hàm này còn tùy thuộc vào câu lệnh SELECT của ngôn ngữ được sử dụng).

*Các toán tử số học:*

^ (lũy thừa), \*(nhân), / (chia), mod (phần dư), +(cộng), - (trừ)

*Các toán tử luận lý:*

not(phủ định), and(phép hội), or (phép tuyển)

*Các toán tử tập hợp:*

In (danh sách các giá trị), LIKE, NOT LIKE, union(phép hợp), intersect (phép giao), minus(phép trừ)

*Các toán tử so sánh :*

=, <>, >, <, >=, <=

*các hàm xử lý ngày tháng*

date()

Trả về ngày tháng năm của hệ thống

time()

Trả về giờ phút giây của hệ thống

day(biểu thức ngày)

Trả về một trị số từ 1 đến 31 của biểu thức ngày

month(biểu thức ngày)

Trả về một số từ 1 đến 12 - là tháng của biểu thức ngày

year(biểu thức ngày)

Trả về năm của biểu thức ngày

len(biểu thức chuỗi)

Trả về chiều dài của chuỗi

*Các hàm tính toán theo nhóm*

sum <thuộc tính>tính tổng giá trị của các bộ theo thuộc tính đã chỉ ra.

max<thuộc tính>:cho biết giá trị lớn nhất của các bộ theo thuộc tính đã chỉ ra.

min<thuộc tính>:cho biết giá trị nhỏ nhất của các bộ theo thuộc tính đã chỉ ra.

avg<thuộc tính>:Cho biết giá trị trung bình của các bộ theo thuộc tính đã chỉ ra.

count \*/ thuộc tính/ distinct <thuộc tính>

count \*: đếm tất cả các bộ

count<thuộc tính>: chỉ đếm những bộ mà giá trị của thuộc tính là khác NULL

count distinct <thuộc tính>

Chỉ đếm những bộ mà giá trị của thuộc tính là khác NULL. hơn nữa, những bộ mà giá trị trùng nhau trên thuộc tính chỉ được đếm là một (đại diện cho cả nhóm).

Sau đây ta sẽ lần lượt tìm hiểu kỹ hơn các mệnh đề của câu lệnh SELECT

Để minh họa cho các ví dụ trong chương này, chúng ta sẽ dùng lại lược đồ cơ sở dữ liệu đã được đề cập trong chương 2.

Sinhvien(MASV, HOTENSV, NU, NGAYSINH, NOISINH, TINH, MALOP)

Lop(MALOP, TENLOP, MAKHOA)

Khoa(MAKHOA, TENKHOA)

Monhoc(MAMH, TENMH, DONVIHT)

Giangvien(MAGV, HOTENGV, HOCVI, CHUYENNGANH, MAKHOA)

Ketqua(MASV, MAMH, LANTHI, DIEMTHI)

Phancong(MALOP, MAMH, MAGV)

### 3.2. TÌM THÔNG TIN TỪ CÁC CỘT CỦA BẢNG - MỆNH ĐỀ SELECT

select distinct /\*/ danh sách thuộc tính/ <biểu thức>, ...

from <danh sách các quan hệ>

-Những thuộc tính được liệt kê trong mệnh đề select sẽ là các thuộc tính có trong quan hệ đích.

-Ký hiệu \* theo sau từ khóa select dùng để chỉ tất cả các thuộc tính của quan hệ nguồn sẽ là thuộc tính của quan hệ đích. Danh sách các thuộc tính cách nhau bởi dấu phẩy và thứ tự này cũng là thứ tự của các thuộc tính trong quan hệ đích.

-Mệnh đề from:

Những quan hệ liên quan đến câu truy vấn được liệt kê sau mệnh đề from, các quan hệ này cách nhau bởi dấu phẩy, thứ tự của các quan hệ được chỉ ra ở đây là không quan trọng.

Cần chú ý rằng khi mệnh đề From chỉ ra từ hai quan hệ trở lên, nếu có một thuộc tính ở mệnh đề select là thuộc tính của nhiều hơn một quan hệ thì cần phải chỉ rõ thuộc tính đó thuộc về quan hệ nào theo cú pháp *tênquanhệ.tênthuộc tính* (sinh viên thường mắc lỗi này khi thực hành với câu lệnh truy vấn SQL)

(Do sinh viên khi học môn này chưa học SQL server, nên nếu khi thực hành bài tập chương này với Access thì cuối mỗi dòng không có dấu chấm phẩy – trừ dòng cuối cùng, ký tự đại diện cho một nhóm ký tự là dấu sao(\*) , còn nếu thực hành với Visual Foxpro thì cuối mỗi dòng có dấu chấm phẩy – trừ dòng cuối cùng)

**Ví dụ 3.1:**

Lập danh sách sinh viên gồm MASV, HOTENSV, NU, NGAYSINH, TINH, MALOP

```
select MASV, HOTENSV, NU, NGAYSINH, TINH, MALOP
from sinhvien;
```

Khi cần lấy thông tin về tất cả các cột của bảng, chúng ta có thể sử dụng dấu sao (\*) thay cho việc liệt kê các tên cột của bảng. Nếu áp dụng cách viết này thì câu lệnh trên tương đương với câu lệnh sau:

```
SELECT *
FROM Sinhvien;
```

(tất nhiên cú pháp này chỉ được sử dụng khi câu truy vấn chỉ liên quan đến một quan hệ)

**Ví dụ 3.2:**

Lập danh sách bao gồm các thông tin về giảng viên như mã số giảng viên, họ và tên giảng viên, học vị, chuyên ngành.

```
SELECT MAGV,HOTENGV,HOCVI, CHUYENNGANH
FROM giangvien;
```

Câu lệnh tìm kiếm thông tin từ các cột của bảng ở trên là cài đặt của phép chiếu trên bốn thuộc tính MAGV,HOTENGV,HOCVI,CHUYENNGANH của quan hệ Giangvien.

Nếu chúng ta muốn đặt tên khác cho tên của các cột của bảng (còn gọi là bí danh- ALIAS), việc này được thực hiện bằng cách thêm từ khóa AS và theo sau là một tên mới. Nếu tên có chứa các ký tự đặc biệt và/hoặc khoảng trắng thì viết tên đó trong cặp dấu ngoặc vuông ( [ ] ).

Chẳng hạn ví dụ 3.2 có thể viết lại là:

```
SELECT MAGV AS [MÃ SỐ GIẢNG VIÊN] ,HOTENGV AS [HỌ VÀ TÊN] HOCVI [TRÌNH ĐỘ] CHUYENNGANH AS [ CHUYÊN NGÀNH]
FROM      Giangvien;
```

Câu lệnh SELECT không chỉ thực hiện việc trích thông tin từ các cột đơn lẻ của bảng mà còn có thể thực hiện các tính toán theo công thức hay biểu thức bất kỳ dựa trên giá trị của các cột trên từng bản ghi của bảng.

Từ khóa DISTINCT nhằm loại bỏ bớt các bộ trùng nhau trong bảng kết quả của lệnh truy vấn (chỉ giữ lại một bộ đại diện cho các bộ giống nhau)

### **Ví dụ 3.3:**

Hãy cho biết các giảng viên của trường thuộc những chuyên ngành nào?

```
SELECT      DISTINCT CHUYENNGANH
FROM      Giangvien;
```

kết quả của câu lệnh này là tất cả những chuyên ngành mà các giảng viên trong trường có thể đảm nhận(tất nhiên mỗi chuyên ngành chỉ xuất hiện một lần trong kết quả truy vấn được).

### **3.3.CHỌN CÁC DÒNG CỦA BẢNG – MỆNH ĐỀ WHERE**

```
SELECT      DISTINCT /*/danh sách thuộc tính/ <biểu thức>,...
FROM      <danh sách các quan hệ>
WHERE      <biểu thức điều kiện>
```

Trong đó <biểu thức điều kiện> có giá trị là hoặc đúng (true) hoặc sai (false). Đây là sự cài đặt của phép chọn trong ngôn ngữ đại số quan hệ.

Nếu điều kiện này chỉ liên quan đến một quan hệ thì gọi là *điều kiện chọn*, nếu điều kiện liên quan đến từ hai quan hệ trở lên thì gọi là *điều kiện kết*. Các điều kiện chọn và điều kiện kết có thể phối hợp với nhau bởi các toán tử logic (and,or,not) để tạo nên những biểu thức logic phức tạp hơn. Cần chú ý rằng thứ tự của các điều kiện ở đây là quan trọng: Nếu có thể thì nên thực hiện *điều kiện chọn* trước khi thực hiện *điều kiện kết* (đây là vấn đề tối ưu hoá câu truy vấn, chúng tôi không đi sâu về vấn đề này[3]).

Sau đây là một số ví dụ cho phép chọn.

**Ví dụ 3.4:**

Lập danh sách những môn học có số đơn vị học trình  $\geq 4$ . Danh sách cần MAMH, TENMH,DONVIHT.

```
SELECT MAMH, TENMH,DONVIHT
FROM monhoc
WHERE DONVIHT  $\geq 4$ ;
```

**Ví dụ 3.5:**

Lập danh sách các sinh viên có mã lớp là CDTH2A, CDTH2B, CDTH2C.

```
SELECT *
FROM sinhvien
WHERE malop="CDTH2A" or malop ="CDTH2B" or malop = "CDTH2C";
```

Cũng có thể viết cách khác như sau:

```
SELECT *
FROM sinhvien
WHERE malop in ("CDTH2A","CDTH2B","CDTH2C");
```

**Ví dụ 3.6:**

Lập danh sách những sinh viên của lớp CDTH2A có điểm thi lần 1 môn CSDL là 6,8

```
Selete masv,diemthi
From ketqua
Where diemthi $\geq 6$  and diemthi $\leq 8$  and mamh="csdl" and lanthi=1;
```

Hoặc có thể viết cách khác

```
SELECT masv,diemthi
FROM ketqua
WHERE diemthi between 6 and 8 and mamh="csdl"
```

Toán tử so sánh tương đối : like

Mẫu so sánh trong phép toán like là một giá trị kiểu text, đó là một dãy ký tự bất kỳ, trong đó có hai ký tự có ý nghĩa đặc biệt sau đây:

? đại diện cho một ký tự bất kỳ tại vị trí có dấu chấm hỏi

% đại diện cho một nhóm ký tự bất kỳ tại vị trí đó

### Ví dụ 3.7:

Lập danh sách các sinh viên có họ là Nguyễn đang theo học tại lớp có mã lớp là CDTH2A

```
SELECT *
FROM sinhvien
WHERE malop="CDTH2A" and HOTENSV like "Nguyễn%";
```

### 3.4.SẮP XẾP CÁC DÒNG CỦA BẢNG - MỆNH ĐỀ ORDER BY

Quan hệ đích có thể được sắp xếp tăng/giảm theo một (hoặc nhiều) thuộc tính nào đó bằng cách sử dụng mệnh đề ORDER BY <danh sách thuộc tính> (độ ưu tiên giảm dần từ trái sang phải), từ khóa DESC (DESCending) được dùng nếu muốn sắp xếp giảm dần, nếu không có DESC, mặc định CSDL sẽ được sắp xếp tăng dần ASC (ASCending) theo các thuộc tính đã chỉ ra.

Nghĩa là danh sách các lớp được sắp xếp theo cột mã khoa, nếu cột mã khoa trùng nhau thì sắp xếp theo cột số học viên

**Sau đây là vấn đề truy vấn thông tin từ nhiều bảng dữ liệu**

### Ví dụ 3.8:

Lập danh sách các lớp có mã khoa là "CNTT" danh sách cần các thông tin MALOP,TENLOP, TENKHOA

```
SELECT MALOP,TENLOP,TENKHOA
FROM lop,khoa
WHERE makhoa="CNTT" and
lop.makhoa=khoa.makhoa;
```



**Ví dụ 3.9:**

Lập danh sách các sinh viên lớp CDTH2A có điểm thi môn học có mã môn học là "869" lớn hơn hay bằng 8.0

```
SELECT Sinhvien.MASV,HOTENSV,NU,NGAYSINH,DIEMTHI
FROM Sinhvien,ketqua
WHERE     malop="CDTH2A" and
          MAMH="869" and
          Sinhvien.MASV=Ketqua.MASV and
          DIEMTHI>=8.0;
```

Cần chú ý rằng do thuộc tính MASV xuất hiện ở cả hai quan hệ Sinhvien và ketqua, nên khi liệt kê nó ở mệnh đề SELECT cần chỉ rõ ra nó thuộc quan hệ nào ? Tuy nhiên sinh viên cũng cần chú ý rằng: nếu ta không ghi Sinhvien.MASV mà ghi là ketqua.MASV thì kết quả vẫn đúng.

**3.5. CÂU LỆNH TRUY VẤN LỒNG NHAU**

Là những câu lệnh mà trong thành phần WHERE có chứa thêm một câu lệnh Select khác nữa. Câu lệnh này thường gặp khi dữ liệu cần thiết phải duyệt qua nhiều lần. Đây là một trong những vấn đề khó khăn nhất khi truy vấn dữ liệu

**Ví dụ 3.10:**

Lập danh sách những sinh viên lớp CDTH2A có điểm thi lần 1 môn học CSDL cao nhất.

Với câu lệnh này nếu dùng các ngôn ngữ lập trình không có ngôn ngữ hỏi cấu trúc thì thật là dài dòng (đầu tiên phải tìm cho ra số điểm lớn nhất thỏa mãn điều kiện trên, sau đó phải duyệt dữ liệu thêm một lần nữa để chọn ra những bộ thỏa đề bài)

```
SELECT     sinhvien.MASV,HOTENSV,NU,NGAYSINH,DIEMTHI
FROM       sinhvien,ketqua
WHERE      MAMH='CSDL' AND
          Lanthi=1
          sinhvien.MASV=Ketqua.MASV AND
```

```
DIEMTHI>=ALL(SELECT DIEMTHI
                FROM ketqua,sinhvien
                WHERE MAMH='CSDL' AND Lanthi=1 and
                sinhvien.MASV=Ketqua.MASV);
```

Tiếp theo sau đây chúng tôi sẽ giới thiệu với bạn đọc thêm một ví dụ về câu lệnh truy vấn lồng nhau:

**Ví dụ 3.11:**

Lập danh sách những giảng viên cùng khoa với giảng viên NGUYEN VAN THANH ?

Giải:

```
select *
from giangvien
where makhoa in
      (select makhoa
       from giangvien
       where Hotengv="NGUYEN VAN THANH");
```

kết quả của câu hỏi con được sử dụng trong phép so sánh với một giá trị khác trong biểu thức điều kiện của câu hỏi bao nó. Các phép so sánh có dạng

```
<phép so sánh>[<lượng từ>](select - câu hỏi con)
```

trong đó phép so sánh có thể là phép so sánh số học hoặc phép so sánh trên tập hợp (chúng tôi đã đề cập ở phần 3.1)

<lượng từ > có thể là ALL,ANY (hoặc SOME). Phép so sánh bằng ANY có thể được thay tương đương bằng phép toán IN, phép so sánh <>ALL có thể thay tương đương bằng phép toán NOT IN.

### 3.6.GOM NHÓM DỮ LIỆU- MỆNH ĐỀ GROUP BY

Khi cần tính toán trên các bộ theo một nhóm nào đó - theo một thuộc tính nào đó, thì ta dùng mệnh đề GROUP BY, chẳng hạn cần tính điểm trung bình chung tất cả các môn học cho tất cả các sinh viên, hay là cần tính số

lượng sinh viên cho mỗi lớp, mỗi khoa, đếm số lượng sinh viên nữ của mỗi khoa, đếm số lượng sinh viên của mỗi tỉnh,...

Mệnh đề GROUP BY <thuộc tính> dùng để phân nhóm dữ liệu. những bộ của bảng có cùng giá trị trên các thuộc tính này sẽ tạo thành một nhóm.

**Ví dụ 3.12:**

Lập bảng điểm trung bình lần 1 các môn học của các sinh viên của lớp có mã lớp là CDTH2A. Danh sách cần: MASV, HOTENSV,DIEMTB( (trong đó DIEMTB là thuộc tính tự đặt).

```
SELECT KETQUA.MASV, HOTENSV,AVG(DIEMTHI) AS DIEMTB
FROM SINHVIEN,KETQUA
WHERE MALOP="CDTH2A" AND LANTHI=1 AND
      SINHVIEN.MASV=KETQUA.MASV
GROUP BY KETQUA.MASV, HOTENSV
```

**Mệnh đề HAVING <điều kiện trên nhóm>**

Nếu cần kiểm tra điều kiện của một nhóm thì dùng mệnh đề Having , chẳng hạn như cho biết những sinh viên nào có điểm trung bình các môn  $\geq 8$ , những khoa nào có nhiều hơn 100 sinh viên nữ,...

Lưu ý những thuộc tính có tham gia vào mệnh đề GROUP BY để phân nhóm phải được liệt kê trong danh sách thuộc tính theo sau từ khóa SELECT.

Mệnh đề HAVING <điều kiện trên nhóm> được sử dụng như là phép chọn phối hợp với việc phân nhóm dữ liệu.

**Ví dụ 3.13:**

Giống như ở ví dụ 3.11 nhưng có thêm điều kiện là điểm trung bình các môn đã thi lớn hơn hoặc bằng 8.0.

```
SELECT KETQUA.MASV, HOTENSV,AVG(DIEMTHI) AS DIEMTB
FROM SINHVIEN,KETQUA,LOP
WHERE MALOP="CDTH2A" AND LANTHI=1 AND
      SINHVIEN.MASV=KETQUA.MASV
GROUP BY KETQUA.MASV, HOTENSV
HAVING AVG(DIEMTHI)>=8.0;
```

Trong một lệnh truy vấn tổng hợp, ngoại trừ thành phần SELECT bắt buộc phải đặt lên đầu, thứ tự của các thành phần khác là tùy ý. Thứ tự dịch một lệnh truy vấn tổng hợp là như sau:

FROM → WHERE → GROUP BY → HAVING → SELECT → ORDER BY

Suy cho cùng, các chương trình quản lý cũng là việc kết xuất các báo cáo từ các quan hệ, mà SQL cho phép tạo ra những quan hệ này một cách tiện lợi. Vì thế hiểu và vận dụng tốt lệnh truy vấn dữ liệu là một việc làm cực kỳ hiệu quả!

**BÀI TẬP****3.1** Dựa vào lược đồ CSDL

Congtrinh(MACT,TENCT,ĐIADIEM,NGAYCAPGP,NGAYKC,NGAYHT)

Nhanvien(MANV,HOTEN,NGAYSINH,PHAI,DIACHI,MAPB)

Phongban(MAPB,TENPB)

Phancong(MACT,MANV,SLNGAYCONG)

Hãy thực hiện các câu hỏi sau bằng SQL

a. Danh sách những nhân viên có tham gia vào công trình có mã công trình(MACT) là X. Yêu cầu các thông tin: MANV, HOTEN, SLNGAYCONG, trong đó MANV được sắp tăng dần

b. Đếm số lượng ngày công của mỗi công trình. Yêu cầu các thông tin: MACT, TENCT, TONGNGAYCONG (TONGNGAYCONG là thuộc tính tự đặt)

c. Danh sách những nhân viên có sinh nhật trong tháng 08. yêu cầu các thông tin: MANV, TENNV, NGAYSINH, DIACHI, TENPB, sắp xếp quan hệ kết quả theo thứ tự tuổi giảm dần.

d. Đếm số lượng nhân viên của mỗi phòng ban. Yêu cầu các thông tin: MAPB, TENPB, SOLUONG. (SOLUONG là thuộc tính tự đặt.)

**3.2.** Dựa vào lược đồ cơ sở dữ liệu

Giaovien(MAGV,HOTEN, MAKHOA)

Monhoc(MAMH,TENMH)

Phonghoc(PHONG,CHUCNANG)

Khoa(MAKHOA,TENKHOA)

Lop(MALOP,TENLOP, MAKHOA)

Lichday(MAGV,MAMH,PHONG,MALOP,NGAYDAY,TUTIET,ĐENTIET, BAIDAY, LYTHUYET, GHICHU)

Hãy thực hiện các câu hỏi sau bằng SQL

a. Xem lịch báo giảng tuần từ ngày 08/09/2003 đến ngày 14/09/2003 của giáo viên có MAGV (mã giáo viên) là TH3A040. Yêu cầu: MAGV, HOTEN, TENLOP, TENMH, PHONG, NGAYDAY, TUTIET, ĐENTIET, BAIDAY, GHICHU)

b. Xem lịch báo giảng ngày 08/09/2003 của các giáo viên có mã khoa là CNTT. Yêu cầu: MAGV, HOTEN, TENLOP, TENMH, PHONG, NGÀYDAY, TUTIET, ĐENTIET, BAIDAY, GHICHU)

c. Cho biết số lượng giáo viên (SOLUONGGV) của mỗi khoa, kết quả cần sắp xếp tăng dần theo cột tên khoa. yêu cầu: TENKHOA, SOLUONGGV (SOLUONGGV là thuộc tính tự đặt)

**3.3.** Hàng năm, Trường X tổ chức kỳ thi giỏi nghề cho các học sinh- sinh viên của trường, mỗi thí sinh sẽ thi hai môn (chẳng hạn các thí sinh thi giỏi nghề công nghệ thông tin thi hai môn là Visual Basic và Cơ Sở Dữ Liệu).

Giả sử lược đồ cơ sở dữ liệu của bài toán quản lý các kỳ thi trên được cho như sau:

THISINH(MASV, HOTEN, NGÀYSINH, MALOP)

LOP(MALOP, TENLOP, MAKHOA)

KHOA(MAKHOA, TENKHOA, ĐIENTHOAI)

MONTHI(MAMT, TENMONTHI)

KETQUA(MASV, MAMT, ĐIỂMTHI)

(Phần giải thích các thuộc tính: HOTEN (họ tên thí sinh), NGÀYSINH (ngày sinh), MALOP (mã lớp), MASV (mã sinh viên), TENLOP (tên lớp), MAKHOA (mã khoa), TENKHOA (tên khoa), ĐIENTHOAI (số điện thoại khoa), MAMT (mã môn thi), TENMONTHI (tên môn thi), ĐIỂMTHI (điểm thi)).

Dựa vào lược đồ cơ sở dữ liệu trên, hãy thực hiện các yêu cầu sau bằng ngôn ngữ

SQL:

a. Hãy cho biết số lượng thí sinh của mỗi khoa đăng ký thi giỏi nghề, cần sắp xếp kết quả theo chiều tăng dần của cột TENKHOA.

b. Lập danh sách những thí sinh đạt danh hiệu giỏi nghề

(Thí sinh đạt danh hiệu giỏi nghề nếu thí sinh không có môn thi nào điểm dưới 8).

c. Lập danh sách những thí sinh nhỏ tuổi nhất có mã khoa là "CNTT" dự thi giỏi nghề.

**3.4.** Cho Lược đồ cơ sở dữ liệu quản lý nhân viên của một công ty như sau:

Nhanvien(MANV, HOTEN, NU, NGÀYSINH, LUONG, MAPB, MACV)

Mỗi nhân viên có một mã nhân viên (MANV) duy nhất, mỗi mã nhân viên xác định họ và tên nhân viên (HOTEN), giới tính (NU), lương (LUONG), mã phòng ban (MAPB), mã chức vụ (MACV).

Phongban(MAPB,TENPB,TRUSO,MANVPHUTRACH,KINHPHI,DOANHTHU)

Mỗi phòng ban có tên gọi phòng ban(TENPB), địa điểm đặt trụ sở (TRUSO), mã nhân viên phụ trách(MANVPHUTRACH), kinh phí hoạt động (KINHPHI), và doanh thu(DOANHTHU)

Chucvu(MACV,TENCV,LUONGTHAPNHAT,LUONGCAONHAT)

Mỗi chức vụ có tên gọi chức vụ (TENCV), mức lương tối thiểu(LUONGTHAPNHAT), mức lương tối đa (LUONGCAONHAT).

Hãy biểu diễn các câu hỏi sau bằng SQL

- a.Lập danh sách gồm các thông tin về các phòng ban trong công ty như: mã số phòng ban, tên phòng ban, địa điểm trụ sở, mã số người phụ trách, kinh phí hoạt động, doanh thu.
- b.Lập danh sách những nhân viên sinh nhật trong tháng 10
- c.Lập danh sách gồm các thông tin mã số nhân viên, họ và tên và lương cả năm của các nhân viên (giả sử rằng lương cả năm =12\*lương)
- d.Lập những phòng ban có kinh phí hoạt động cao nhất.
- e.Lập danh sách nhân viên của phòng ban có mã số phòng ban là 40.
- f.Lập danh sách nhân viên của phòng có mã số phòng ban 10,30,50.
- g.Lập danh sách các nhân viên có lương tháng từ 2.500.000 đến 4.000.000
- h.Tìm những nhân viên có tuổi cao nhất thuộc phòng ban có MAPB là 10
- i.Lập danh sách các nhân viên của phòng 10,30,50. kết quả in ra theo thứ tự tăng dần của mã phòng nếu trùng mã phòng thì sắp xếp giảm dần theo mức lương.
- k.Lập danh sách các nhân viên phòng 10,30,50, chỉ in ra những người là lãnh đạo của mỗi phòng ban này.
- l.lập danh sách gồm mã phòng mà người có mức lương cao nhất của phòng lớn hơn hoặc bằng 4.000.000
- m.Lập mã phòng ban, tên phòng ban, họ và tên của lãnh đạo phòng tương ứng.
- n.Lập danh sách những người làm việc cùng phòng với ông Nguyen Van Thanh
- o.Lập biết mã số nhân viên, họ và tên, mức lương của người lãnh đạo ông Nguyen Van Thanh.
- p.Lập danh sách nhân viên có mức lương lớn hơn hay bằng mức lương cao nhất của phòng ông Nguyen Van Thanh.

**q.** Cho biết mã số nhân viên, họ và tên, tổng số nhân viên, mức lương cao nhất, mức lương thấp nhất, mức lương trung bình của từng phòng ban.

**r.** Cho biết các nhân viên có mức lương cao nhất của các phòng ban.

**s.** Cho biết số lượng nhân viên của mỗi phòng ban.



## chương 4

**RÀNG BUỘC TOÀN VỆ**

(Integrity Constraint)

**4.1 RÀNG BUỘC TOÀN VỆ****4.1.1 Khái Niệm Ràng Buộc Toàn Vệ**

Trong mỗi CSDL luôn tồn tại nhiều mối liên hệ giữa các thuộc tính, giữa các bộ; sự liên hệ này có thể xảy ra trong cùng một quan hệ hoặc trong các quan hệ của một lược đồ CSDL. Các mối liên hệ này là những điều kiện bất biến mà tất cả các bộ của những quan hệ có liên quan trong CSDL đều phải thoả mãn ở mọi thời điểm. Những điều kiện bất biến đó được gọi là *ràng buộc toàn vệ*.. Trong thực tế ràng buộc toàn vệ là các quy tắc quản lý được áp đặt trên các đối tượng của thế giới thực. Chẳng hạn mỗi sinh viên phải có một mã sinh viên duy nhất, hai thí sinh dự thi vào một trường phải có số báo danh khác nhau, một sinh viên dự thi một môn học không quá 3 lần,...

Nhiệm vụ của người phân tích thiết kế là phải phát hiện càng đầy đủ các ràng buộc toàn vệ càng tốt và mô tả chúng một cách chính xác trong hồ sơ phân tích thiết kế - đó là một việc làm rất quan trọng. Ràng buộc toàn vệ được xem như là một công cụ để diễn đạt ngữ nghĩa của CSDL. Một CSDL được thiết kế công kênh nhưng nó thể hiện được đầy đủ ngữ nghĩa của thực tế vẫn có giá trị cao hơn rất nhiều so với một cách thiết kế gọn nhẹ nhưng nghèo nàn về ngữ nghĩa vì thiếu các ràng buộc toàn vệ của CSDL.

Công việc kiểm tra ràng buộc toàn vệ thường được tiến hành vào thời điểm cập nhật dữ liệu ( thêm, sửa, xoá). Những ràng buộc toàn vệ phát sinh phải cần được ghi nhận và xử lý một cách tường minh (thường là bởi một hàm chuẩn hoặc một đoạn chương trình).

Ràng buộc toàn vệ và kiểm tra sự vi phạm ràng buộc toàn vệ là hai trong số những vấn đề quan trọng trong quá trình phân tích thiết kế cơ sở dữ liệu, nếu không quan tâm đúng mức đến những vấn đề trên, thì có thể dẫn đến

những hậu quả nghiêm trọng về tính an toàn và toàn vẹn dữ liệu , đặc biệt là đối với những cơ sở dữ liệu lớn.

#### **4.1.2 Các Yếu Tố Của Ràng Buộc Toàn Vẹn**

Mỗi ràng buộc toàn vẹn có bốn yếu tố: điều kiện, bối cảnh, bảng tầm ảnh hưởng và hành động phải cần thực hiện khi phát hiện có ràng buộc toàn vẹn bị vi phạm:

##### **4.1.2.1.Điều kiện**

Điều kiện của ràng buộc toàn vẹn là sự mô tả, và biểu diễn hình thức nội dung của nó

Điều kiện của một ràng buộc toàn vẹn R có thể được biểu diễn bằng ngôn ngữ tự nhiên, ngôn ngữ đại số quan hệ, ngôn ngữ mã giả, ngôn ngữ truy vấn SQL,... ngoài ra điều kiện của ràng buộc toàn vẹn cũng có thể được biểu diễn bằng phụ thuộc hàm (khái niệm phụ thuộc hàm sẽ được đề cập trong chương 5)

Sau đây là một số ràng buộc toàn vẹn trên lược đồ CSDL quản lý sinh viên .

Mỗi lớp học phải có một mã số duy nhất để phân biệt với các lớp học khác trong trường.

Mỗi lớp học phải thuộc về một khoa của trường.

Mỗi sinh viên có một mã số sinh viên duy nhất, không trùng với bất cứ sinh viên nào trong trường.

Mỗi học viên phải đăng ký vào một lớp học trong trường.

Mỗi học viên chỉ được thi tối đa 3 lần cho mỗi môn học.

Tổng số học viên của một lớp phải lớn hơn hoặc bằng số lượng đếm được của một lớp tại một thời điểm nào đó.

##### **4.1.2.2.Bối cảnh**

Bối cảnh của ràng buộc toàn vẹn là những quan hệ mà ràng buộc đó có hiệu lực hay nói một cách khác, đó là những quan hệ cần phải được kiểm tra khi tiến hành cập nhật dữ liệu. Bối cảnh của một ràng buộc toàn vẹn có thể là một hoặc nhiều quan hệ.

Chẳng hạn với ràng buộc toàn vẹn R trên thì bối cảnh của nó là quan hệ Sinhvien

#### 4.1.2.3. Bảng tầm ảnh hưởng

Trong quá trình phân tích thiết kế một CSDL, người phân tích cần lập bảng tầm ảnh hưởng cho một ràng buộc toàn vẹn nhằm xác định thời điểm cần phải tiến hành kiểm tra khi tiến hành cập nhật dữ liệu.

Thời điểm cần phải kiểm tra ràng buộc toàn vẹn chính là thời điểm cập nhật dữ liệu.

Một bảng tầm ảnh hưởng của một ràng buộc toàn vẹn có dạng sau:

Tên RBTV	Thêm(T)	Sửa(S)	Xoá(X)
$r_1$	+		
$r_2$		-	
$r_3$			-(*)
$r_n$			

Bảng này chứa toàn các ký hiệu + , - hoặc -(\*).

Chẳng hạn + tại (dòng  $r_1$ , cột Thêm) thì có nghĩa là khi thêm một bộ vào quan hệ  $r_1$  thì RBTV bị vi phạm.

Dấu - Tại ô (dòng  $r_2$ , cột sửa) thì có nghĩa là khi sửa một bộ trên quan hệ  $r_2$  thì RBTV không bị vi phạm.

, ...

*Quy ước:*

-Không được sửa thuộc tính khoá.

-Nếu không bị vi phạm do không được phép sửa đổi thì ký hiệu là -(\*).

#### 4.1.2.4. Hành động cần phải có khi phát hiện có RBTV bị vi phạm:

khi một ràng buộc toàn vẹn bị vi phạm, cần có những hành động thích hợp. Thông thường có 2 giải pháp:

Thứ nhất: Đưa ra thông báo và yêu cầu sửa chữa dữ liệu của các thuộc tính cho phù hợp với quy tắc đảm bảo tính nhất quán dữ liệu. Thông báo phải

đầy đủ và phải thân thiện với người sử dụng. Giải pháp này là phù hợp cho việc xử lý thời gian thực.

Thứ hai: Từ chối thao tác cập nhật. Giải pháp này là phù hợp đối với việc xử lý theo lô. Việc từ chối cũng phải được lưu lại bằng những thông báo đầy đủ, rõ ràng vì sao thao tác bị từ chối và cần phải sửa lại những dữ liệu nào ?

Khóa nội, khóa ngoại, giá trị NOT NULL là những ràng buộc toàn vẹn miền giá trị của các thuộc tính. Những ràng buộc toàn vẹn này là những ràng buộc toàn vẹn đơn giản trong CSDL.

Các hệ quản trị cơ sở dữ liệu thường có các cơ chế tự động kiểm tra các ràng buộc toàn vẹn về miền giá trị của khóa nội, khóa ngoại, giá trị NOT NULL.

Việc kiểm tra ràng buộc toàn vẹn có thể tiến hành vào những thời điểm sau đây.

Thứ nhất: Kiểm tra ngay sau khi thực hiện một thao tác cập nhật CSDL. Thao tác cập nhật chỉ được xem là hợp lệ nếu như nó không vi phạm bất cứ một ràng buộc toàn vẹn nào, nghĩa là nó không làm mất tính toàn vẹn của CSDL. Nếu vi phạm ràng buộc toàn vẹn, thao tác cập nhật bị coi là không hợp lệ và sẽ bị hệ thống huỷ bỏ (hoặc có một xử lý thích hợp nào đó)

Thứ hai: Kiểm tra định kỳ hay đột xuất, nghĩa là việc kiểm tra ràng buộc toàn vẹn được tiến hành độc lập với thao tác cập nhật dữ liệu. Đối với những trường hợp vi phạm ràng buộc toàn vẹn, hệ thống có những xử lý ngầm định hoặc yêu cầu người sử dụng xử lý những sai sót một cách tường minh.

## **4.2. PHÂN LOẠI RÀNG BUỘC TOÀN VẸN**

Trong quá trình phân tích thiết kế CSDL, người phân tích phải phát hiện tất cả các ràng buộc toàn vẹn tiềm ẩn trong CSDL đó. Việc phân loại các ràng buộc toàn vẹn là rất có ích, nó nhằm giúp cho người phân tích có được một định hướng để phát hiện các ràng buộc toàn vẹn, tránh bỏ sót. Các ràng buộc toàn vẹn có thể được chia làm hai loại chính như sau:

Thứ nhất: Ràng buộc toàn vẹn có phạm vi là một quan hệ bao gồm :Ràng buộc toàn vẹn miền giá trị, ràng buộc toàn vẹn liên thuộc tính, ràng buộc toàn vẹn liên bộ.

Thứ hai: Ràng buộc toàn vẹn có phạm vi là nhiều quan hệ bao gồm :Ràng buộc toàn vẹn phụ thuộc tồn tại, ràng buộc toàn vẹn liên bộ - liên quan hệ, ràng buộc toàn vẹn liên thuộc tính - liên quan hệ.

Để minh họa cho phần lý thuyết của chương này, chúng ta xét ví dụ sau đây:

#### **Ví dụ 4.1**

Cho một CSDL C dùng để quản lý việc đặt hàng và giao hàng của một công ty. Lược đồ CSDL C gồm các lược đồ quan hệ như sau:

Q<sub>1</sub>: Khách (MAKH, TENKH, DIACHIKH, DIENTHOAI)

*Tên từ:*

Mỗi khách hàng có một mã khách hàng (MAKH) duy nhất, mỗi MAKH xác định tên khách hàng (TENKH), địa chỉ (DIACHIKH), số điện thoại (DIENTHOAI).

Q<sub>2</sub>: Hang(MAHANG, TENHANG, QUYCACH, DVTINH)

*Tên từ:*

Mỗi mặt hàng có một mã hàng (MAHANG) duy nhất, mỗi MAHANG xác định tên hàng (TENHANG), quy cách hàng (QUYCACH), đơn vị tính (DVTINH).

Q<sub>3</sub>: Dathang(SODH, MAHANG, SLDAT, NGAYDH, MAKH)

*Tên từ:*

Mỗi mã số đặt hàng (SODH) xác định một ngày đặt hàng (NGAYDH) và mã khách hàng tương ứng (MAKH). Biết mã số đặt hàng và mã mặt hàng thì biết được số lượng đặt hàng (SLDAT). Mỗi khách hàng trong một ngày có thể có nhiều lần đặt hàng

Q<sub>4</sub>: Hoadon(SOHD, NGAYLAP, SODH, TRIGIAHD, NGAYXUAT)

*Tên từ:*

Mỗi hoá đơn tổng hợp có một mã số duy nhất là SOHD, mỗi hoá đơn bán hàng có thể gồm nhiều mặt hàng. Mỗi hoá đơn xác định ngày lập hoá đơn

(NGAYLAP), ứng với số đặt hàng nào (SODH). Giả sử rằng hoá đơn bán hàng theo yêu cầu của chỉ một đơn đặt hàng có mã số là SODH và ngược lại, mỗi đơn đặt hàng chỉ được giải quyết chỉ trong một hoá đơn. Do điều kiện khách quan có thể công ty không giao đầy đủ các mặt hàng cũng như số lượng từng mặt hàng như yêu cầu trong đơn đặt hàng nhưng không bao giờ giao vượt ngoài yêu cầu. Mỗi hóa đơn xác định một trị giá của nhưng các mặt hàng trong hoá đơn (TRIGIAHD) và một ngày xuất kho giao hàng cho khách (NGAYXUAT)

Q<sub>5</sub>: Chitiethd (SOHD, MAHANG, GIABAN, SLBAN)

*Tân từ:*

Mỗi SOHD, MAHANG xác định giá bán (GIABAN) và số lượng bán (SLBAN) của một mặt hàng trong một hoá đơn.

Q<sub>6</sub>: Phieuthu(SOPT, NGAYTHU, MAKH, SOTIEN)

*Tân từ:*

Mỗi phiếu thu có một số phiếu thu (SOPT) duy nhất, mỗi SOPT xác định một ngày thu (NGAYTHU) của một khách hàng có mã khách hàng là MAKH và số tiền thu là SOTIEN. Mỗi khách hàng trong một ngày có thể có nhiều số phiếu thu.

#### 4.2.1. Ràng buộc toàn vẹn có bối cảnh là một quan hệ

4.2.1.1. Ràng Buộc Toàn Vẹn liên bộ:

+Ràng buộc toàn vẹn về khoá chính:

Đây là một trường hợp đặc biệt của Ràng Buộc toàn Vẹn liên bộ, RBTV này rất phổ biến và thường được các hệ quản trị CSDL tự động kiểm tra.

#### Ví dụ 4.2:

Với r là một quan hệ trên lược đồ quan hệ Khách ta có ràng buộc toàn vẹn sau:

$R_1: \forall t_1, t_2 \in r$

$t_1. MAKH \neq t_2. MAKH$

Cuối  $\forall$

R <sub>1</sub>	Thêm	Sửa	Xoá
Khach	+	-	-

+Ràng buộc toàn vẹn về tính duy nhất

Ví dụ: mỗi phòng ban phải có một tên gọi duy nhất

+Ngoài ra nhiều khi ta còn gặp những RBTV khác chẳng hạn như RBTV sau trong quan hệ sau đây.

Ví dụ: KETQUA(MASV, MAMH, LANTHI, DIEM)

Mỗi sinh viên chỉ được đăng thi mỗi môn tối đa là 3 lần.

#### 4.2.1.2. Ràng Buộc Toàn Vẹn Về Miền Giá Trị

Ràng buộc toàn vẹn có liên quan đến miền giá trị của các thuộc tính trong một quan hệ. Ràng buộc này thường gặp. Thông thường các hệ quản trị CSDL đã tự động kiểm tra (một số) ràng buộc loại này.

#### Ví dụ 4.3:

Với r là một quan hệ của Hoadon ta có ràng buộc toàn vẹn sau

$R_3: \forall t \in r$

$t.TRIGIAHD > 0$

Cuối  $\forall$

$R_3$	Thêm	Sửa	Xoá
Hoadon	+	+	-

#### 4.2.1.3. Ràng Buộc Toàn Vẹn Liên Thuộc Tính

Ràng buộc toàn vẹn liên thuộc tính (một quan hệ) là mối liên hệ giữa các thuộc tính trong một lược đồ quan hệ.

#### Ví dụ 4.4

Với r là một quan hệ của Hoadon ta có ràng buộc toàn vẹn sau:

$R_4: \forall t \in r$

$t.NGAYLAP \leq t.NGAYXUAT$

Cuối  $\forall$

$R_4$	Thêm	Sửa	Xoá
Hoadon	+	+	-

#### 4.2.2. Ràng buộc toàn vẹn có bối cảnh là nhiều quan hệ

##### 4.2.2.1. Ràng Buộc Toàn Vẹn Về Khoá Ngoại:

Ràng buộc toàn vẹn về khoá ngoại còn được gọi là ràng buộc toàn vẹn phụ thuộc tồn tại. Cũng giống như ràng buộc toàn vẹn về khoá nội, loại ràng buộc toàn vẹn này rất phổ biến trong các CSDL.

#### Ví dụ 4.5

$R_2$ . dathang[MAKH]  $\subseteq$  khách[MAKH]

$R_2$	Thêm	Sửa	Xoá
dathang	+	+	-
Khach	-	-	+

#### 4.2.2.2. Ràng Buộc Toàn Vẹn Liên Thuộc Tính Liên Quan Hệ

Ràng buộc loại này là mối liên hệ giữa các thuộc tính trong nhiều lược đồ quan hệ.

#### Ví dụ 4.6

Với  $r, s$  lần lượt là quan hệ của Dathang và Hoadon. Ta có ràng buộc toàn vẹn  $R_5$  như sau:

$R_5: \forall t_1 \in r, t_2 \in s$

Nếu  $t_1.SODH = t_2.SODH$  thì

$t_1.NGAYDH \leq t_2.NGAYXUAT$

Cuối  $\forall$

$R_5$	Thêm	Sửa	Xoá
Dathang	+	-	-
Hoandon	+	+	-

#### 4.2.2.3. Ràng Buộc Toàn Vẹn Liên Bộ Liên Quan Hệ

Ràng buộc loại này là mối liên hệ giữa các bộ trong một lược đồ cơ sở dữ liệu. Chẳng hạn như tổng số tiền phải trả trong mỗi hoá đơn (chitiethd) phải bằng TRỊ GIÁ HOÁ ĐƠN của hoá đơn đó trong quan hệ Hoadon. Hoặc số lượng học viên trong một lớp phải bằng SOHOCVIEN của lớp đó.

Ngoài ra còn có một số loại RBTV khác như :RBTV về thuộc tính tổng hợp, RBTV do tồn tại chu trình, RBTV về giá trị thuộc tính theo thời gian.



## BÀI TẬP

**4.1.** Việc tổ chức kỳ thi tốt nghiệp của một khoa như sau:

Mỗi thí sinh có một Mã số sinh viên duy nhất (MASV), mỗi MASV xác định được các thông tin: họ và tên (HOTEN), ngày sinh (NGAYSINH), nơi sinh, nữ,phái, dân tộc.

Mỗi lớp có một mã lớp (MALOP) duy nhất , mỗi mã lớp xác định các thông tin: tên lớp (TENLOP), mỗi lớp chỉ thuộc sự quản lý của một khoa nào đó. Mỗi khoa có một mã khoa duy nhất (MAKHOA), mỗi mã khoa xác định tên khoa (TENKHOA).

Mỗi thí sinh đều phải dự thi tốt nghiệp ba môn. Mỗi môn thi có một mã môn thi (MAMT) duy nhất, mỗi mã môn thi xác định các thông tin: tên môn thi (TENMT), thời gian làm bài – được tính bằng phút (PHUT), ngày thi (NGAYTHI), buổi thi (BUOITHI), môn thi này là môn lý thuyết hay thực hành (LYTHUYET). Chú ý rằng, nếu một môn học được cho thi ở nhiều hệ thì được đặt MAMT khác nhau (chẳng hạn cả trung cấp và cao đẳng ngành công nghệ thông tin đều thi môn Cơ Sở Dữ Liệu), để diễn tả điều này, mỗi mã môn học cần phải được ghi chú (GHICHU) để cho biết môn thi đó dành cho khối nào trung cấp, hay cao đẳng). Mỗi thí sinh ứng với một môn thi có một điểm thi (DIEMTHI) duy nhất, điểm thi được chấm theo thang điểm 10 và có lấy điểm lẻ đến 0.5. Một thí sinh được coi là đậu tốt nghiệp nếu điểm thi của tất cả các môn của thí sinh đó đều lớn hơn hoặc bằng 5.

Trong một phòng thi có thể có thí sinh của nhiều lớp. Trong một kỳ thi, mỗi thí sinh có thể thi tại những phòng thi (PHONGTHI) khác nhau, chẳng hạn một thí sinh thi tốt nghiệp ba môn là Cơ sở dữ liệu, Lập trình C và Visual Basic thì môn Cơ Sở Dữ Liệu và Lập Trình C thi tại phòng F6.1, còn môn thực hành Visual Basic thi tại phòng máy H7.3

Qua phân tích sơ bộ trên, ta có thể lập một lược đồ cơ sở dữ liệu như sau:

THISINH(MASV,HOTEN,NGAYSINH,MALOP)

LOP(MALOP,TENLOP)

MONTHI(MAMT,TENMT, LYTHUYET,PHUT,NGAYTHI,BUOITHI,GHICHU)

KETQUA(MASV,MAMT,DIEMTHI)

a.Tìm khoá cho mỗi lược đồ quan hệ trên.

b.Hãy phát biểu các ràng buộc toàn có trong cơ sở dữ liệu trên.

**4.2.** Cho lược đồ cơ sở dữ liệu (đã được phân tích ở Ví dụ 2.1)

Hãy phát biểu các ràng buộc toàn có trong lược đồ cơ sở dữ liệu trên.

**4.3.** Cho lược đồ cơ sở dữ liệu ở bài tập 4.1. Thực hiện các yêu cầu sau bằng ngôn ngữ SQL:

a. Lập bảng điểm môn thi có mã môn thi là “CSDL02” cho tất các thí sinh có mã lớp là “CDTH2A”. danh sách cần MASV, HOTEN, NGAYSINH, DIEMTHI và được sắp xếp tăng dần theo MASV.

b. Hãy thống kê xem mỗi môn thi có bao nhiêu thí sinh có điểm thi lớn hơn hay bằng 5? Danh sách cần: MAMT, TENMT, GHICHU, SOLUONG trong đó số lượng (SOLUONG) là thuộc tính tự đặt.

c. Lập danh sách những thí sinh đậu tốt nghiệp (theo tiêu chuẩn đã phân tích ở trên), danh sách cần: MASV, HOTEN, NGAYSINH, DIEMTONG, trong đó DIEMTONG bằng tổng điểm thi của 3 môn thi, DIEMTONG là thuộc tính tự đặt.

d. Nếu cần mở rộng bài toán theo hai hướng; Thứ nhất là quản lý kỳ thi tốt nghiệp cho tất cả các khoa trong toàn trường, Thứ hai là quản lý thông tin về phòng thi (PHONGTHI) của mỗi thí sinh, thì lược đồ cơ sở dữ liệu trên cần phải được điều chỉnh như thế nào ?

**4.4.** Hãy tìm các ràng buộc toàn vẹn có trong mỗi lược đồ cơ sở dữ liệu ở các bài tập

3.1. đến 3.4.

chương 5

## LÝ THUYẾT THIẾT KẾ CƠ SỞ DỮ LIỆU

### 5.1. CÁC VẤN ĐỀ GẶP PHẢI KHI TỔ CHỨC DỮ LIỆU:

Trước khi bàn về cách thiết kế một cơ sở dữ liệu tốt, chúng ta hãy phân tích xem tại sao trong một số lược đồ quan hệ lại tồn tại những vấn đề rắc rối. Chẳng hạn cho lược đồ quan hệ:

Thi(MASV,HOTEN,MONHOC,DIEMTHI)

và sau đây là một quan hệ trên lược đồ quan hệ Thi

MASV	HOTEN	MONHOC	DIEMTHI
00CDTH189	Nguyễn Văn Thành	Cấu Trúc Dữ Liệu	7
00CDTH189	Nguyễn Văn Thành	Cơ Sở Dữ Liệu	9
00CDTH211	Trần Thu Hà	Kỹ Thuật Lập Trình	5
00CDTH189	Nguyễn Văn Thành	Kỹ Thuật Lập Trình	8

Quan hệ này ghi kết quả điểm thi các môn của các sinh viên. Chúng ta có thể nhận thấy một số vấn đề nảy sinh sau:

1)Dư thừa (redundancy): Họ tên của các sinh viên được lặp lại mỗi lần cho mỗi môn thi.

2)Mâu thuẫn tiềm ẩn (potentia inconsistancy) hay bất thường khi cập nhật. Do hậu quả của dư thừa, chúng ta có thể cập nhật họ tên của một sinh viên trong một bộ nào đó nhưng vẫn để lại họ tên cũ trong những bộ khác. Vì vậy chúng ta có thể không có một họ tên duy nhất đối với mỗi sinh viên như chúng ta mong muốn.

3)Bất thường khi chèn (insertion anomaly). Chúng ta không thể biết họ tên của một sinh viên nếu hiện tại sinh viên đó không dự thi môn nào.

4)Bất thường khi xóa (deletion anomaly). Ngược lại với vấn đề 3) là vấn đề chúng ta có thể xóa tất cả các môn thi của một sinh viên, vô ý làm mất dấu vết để tìm ra họ tên của sinh viên này.

Những vấn đề nêu trên sẽ được giải quyết nếu chúng ta phân rã lược đồ quan hệ Diemthi thành hai lược đồ quan hệ:

Sinhvien(MASV,HOTEN)

Ketqua(MASV,MONHOC,DIEMTHI)

Lúc này lược đồ quan hệ Sinhvien cho biết họ tên của mỗi sinh viên chỉ xuất hiện đúng một lần; do vậy không có dư thừa. Ngoài ra chúng ta cũng có thể nhập họ tên của một sinh viên dù hiện tại sinh viên đó chưa có kết quả thi môn nào. Tuy nhiên lúc này ta nhận thấy rằng để tìm danh sách họ tên của các sinh viên ứng với môn thi cơ sở dữ liệu thì chúng ta phải thực hiện một phép kết nối, còn với một quan hệ duy nhất thì chúng ta có thể dễ dàng trả lời bằng cách thực hiện một phép chọn rồi một phép chiếu. Làm sao để đưa được một lược đồ cơ sở dữ liệu chưa tốt về một lược đồ cơ sở dữ liệu tốt hơn? chương này và chương tới nhằm giải quyết vấn đề này.

## 5.2. PHỤ THUỘC HÀM

Phụ thuộc hàm (functional dependancy) là một công cụ dùng để biểu diễn một cách hình thức các ràng buộc toàn vẹn. Phương pháp biểu diễn này có rất nhiều ưu điểm, và đây là một công cụ kỳ quan trọng, gắn chặt với lý thuyết thiết kế cơ sở dữ liệu.

Trong chương này chúng ta sẽ tìm hiểu về lý thuyết thiết kế cơ sở dữ liệu quan hệ, mà bắt đầu là phụ thuộc hàm và một số ứng dụng trong việc giải quyết các bài toán như: tìm khoá, tìm phủ tối thiểu, xác định dạng chuẩn. Trong chương tới chúng ta sẽ tiếp tục tìm hiểu về cách thức chuẩn hoá một cơ sở dữ liệu.

### 5.2.1 Định Nghĩa Phụ Thuộc Hàm

Cho lược đồ quan hệ  $Q\{A_1, A_2, \dots, A_n\}$ .  $X, Y$  là hai tập con khác rỗng của  $Q^+$ . Ta nói  $X$  xác định  $Y$  (hay  $Y$  phụ thuộc hàm vào  $X$ ) nếu với  $r$  là một quan hệ nào đó trên  $Q$ ,  $\forall t_1, t_2 \in r$  mà  $t_1.X = t_2.X \Rightarrow t_1.Y = t_2.Y$  (nghĩa là không thể tồn tại hai bộ trong  $r$  giống nhau ở các thuộc tính trong tập  $X$  mà lại khác nhau ở một hay nhiều thuộc tính nào đó trong tập  $Y$ ). Khi đó ta ký hiệu là  $X \rightarrow Y$ .

Chẳng hạn như phụ thuộc hàm của thuộc tính họ tên của sinh viên (HOTENSV) vào mã số sinh viên (MASV) và ta có thể diễn tả bằng phụ thuộc hàm:

MASV  $\rightarrow$  HOTENSV

Phụ thuộc hàm  $X \rightarrow X$  được gọi là *phụ thuộc hàm hiển nhiên*. người ta thường dùng F để chỉ tập các phụ thuộc hàm định nghĩa trên Q. Vì Q hữu hạn nên F cũng hữu hạn, ta có thể đánh số các phụ thuộc hàm của F là  $f_1, f_2, \dots, f_m$ .

Quy ước: chỉ cần mô tả các phụ thuộc hàm không hiển nhiên trong tập F, các phụ thuộc hàm hiển nhiên được ngầm hiểu là đã có trong F.

### Ví dụ 5.1:

Cho lược đồ quan hệ Q(ABCDE), r là quan hệ xác định trên Q được cho như sau:

A	B	C	D	E
a1	b1	c1	d1	e1
a1	b2	c2	d2	e1
a2	b1	c3	d3	e1
a2	b1	c4	d3	e1
a3	b2	c3	d1	e1

Những phụ thuộc hàm nào sau đây thoả r ?

$A \rightarrow D$ ;  $AB \rightarrow D$ ;  $E \rightarrow A$ ;  $A \rightarrow E$ ;

Giải:

$AB \rightarrow D$ ;  $A \rightarrow E$ ;

### 5.2.2 Cách Xác Định Phụ Thuộc Hàm Cho Lược Đồ Quan Hệ

Cách duy nhất để xác định đúng các phụ thuộc thích hợp cho một lược đồ quan hệ là xem xét nội dung tân từ của lược đồ quan hệ đó.

Chẳng hạn với lược đồ cơ sở dữ liệu đã cho trong ví dụ 2.1, thì phụ thuộc hàm ứng với từng lược đồ quan hệ được xác định như sau:

MASV  $\rightarrow$  HOTENSV, NU, NGAYSINH, MALOP, TINH

MALOP  $\rightarrow$  TENLOP, MAKHOA

MAKHOA  $\rightarrow$  TENKHOA

MAMH  $\rightarrow$  TENMH, DONVIHT

MASV, MAMH, LANTHI  $\rightarrow$  DIEMTHI

.....

### 5.2.3 Một Số Tính Chất Của Phụ Thuộc Hàm - hệ luật dẫn Armstrong

Để có thể xác định được các phụ thuộc hàm khác từ tập phụ thuộc hàm đã có, ta dùng hệ tiên đề Armstrong (1974), gồm các luật sau:

với  $X, Y, Z, W \subseteq Q^+$

1. *Luật phản xạ* (reflexivity)

$$X \supseteq Y \Rightarrow X \rightarrow Y$$

Quy tắc này đưa ra những phụ thuộc hàm hiển nhiên (phụ thuộc hàm tầm thường), đó là những phụ thuộc hàm mà vế trái bao hàm cả vế phải. Những phụ thuộc hàm hiển nhiên đều đúng trong mọi quan hệ.

2. *Luật tăng trưởng* (augmentation)

$$X \rightarrow Y \Rightarrow XZ \rightarrow YZ$$

3. *Luật bắc cầu* (transitivity)

$$X \rightarrow Y, Y \rightarrow Z \Rightarrow X \rightarrow Z$$

*Các quy tắc suy rộng:*

4. *Luật hợp* (the union rule)

$$\text{Cho } X \rightarrow Y, X \rightarrow Z \Rightarrow X \rightarrow YZ$$

5. *Luật bắc cầu giả* (the pseudotransitivity rule)

$$\text{Cho } X \rightarrow Y, WY \rightarrow Z \Rightarrow XW \rightarrow Z$$

6. *Luật phân rã* (the decomposition rule):

$$\text{Cho } X \rightarrow Y, Z \subseteq Y \Rightarrow X \rightarrow Z$$

## 5.3 BAO ĐÓNG CỦA TẬP PHỤ THUỘC HÀM VÀ BAO ĐÓNG CỦA TẬP THUỘC TÍNH

### 5.3.1. Bao Đóng Của Tập Phụ Thuộc Hàm F

*Bao đóng (closure) của tập phụ thuộc hàm F* (ký hiệu là  $F^+$ ) là tập hợp tất cả các phụ thuộc hàm có thể suy ra từ F dựa vào các tiên đề Armstrong. Rõ ràng  $F \subseteq F^+$

#### Ví dụ 5.2

Cho lược đồ quan hệ Q(ABCDEFGH) và F được cho như sau:

$$F = \{B \rightarrow A; DA \rightarrow CE; D \rightarrow H; GH \rightarrow C; AC \rightarrow D\}$$

$$\text{Khi đó } F^+ = \{B \rightarrow A; DA \rightarrow CE; D \rightarrow H; GH \rightarrow C; AC \rightarrow D;$$

$$BC \rightarrow AC; BC \rightarrow D; DA \rightarrow AH; DG \rightarrow C; BC \rightarrow AD; \dots\}$$

(Lưu ý rằng, nếu mỗi thuộc tính được biểu diễn bằng một ký tự thì danh sách các thuộc tính có hoặc không có dấu phẩy đều được, còn giữa các phụ thuộc hàm phải có dấu chấm phẩy)

Các tính chất của tập  $F^+$

1. *Tính phản xạ:*

Với mọi tập phụ thuộc hàm  $F^+$  ta luôn có  $F \subseteq F^+$

2. *Tính đơn điệu:*

Nếu  $F \subseteq G$  thì  $F^+ \subseteq G^+$

3. *Tính lũy đẳng:*

Với mọi tập phụ thuộc hàm  $F$  ta luôn luôn có  $F^{++} = F^+$ .

### 5.3.2. Bao Đóng Của Tập Thuộc Tính X

Cho lược đồ quan hệ  $Q$ . giả sử  $F$  là tập các phụ thuộc hàm trong  $Q$ ,  $X \subseteq Q^+$ .

Bao đóng của tập thuộc tính  $X$  đối với  $F$  ký hiệu là  $X^+$  (hoặc  $X_F^+$ ) là tập tất cả các thuộc tính  $A \in Q^+$  được suy ra từ  $X$  dựa vào các phụ thuộc hàm trong  $F$  và hệ tiên đề Armstrong, nghĩa là:

$$X^+ = \{A : A \in Q^+ \text{ và } X \rightarrow A \in F^+\}$$

#### Ví dụ 5.3

Cho lược đồ quan hệ  $Q(ABCDEFGH)$  và tập phụ thuộc hàm  $F$

$$F = \{B \rightarrow A; DA \rightarrow CE; D \rightarrow H; GH \rightarrow C; AC \rightarrow D\}$$

Hãy tính:

$$B^+; H^+; BC^+$$

Giải

Khi đó  $B^+ = BA$  ; (do có phụ thuộc hàm  $B \rightarrow A$ )

$H^+ = H$ . (do có phụ thuộc hàm  $H \rightarrow H$ )

$BC^+ = BCAD E H$ . (do có các phụ thuộc hàm:  $B \rightarrow A; AC \rightarrow D; DA \rightarrow CE; D \rightarrow H$ )

Tương tự như tập bao đóng của tập phụ thuộc hàm  $F^+$ , tập bao đóng  $X^+$  cũng chứa các phần tử của tập  $X$ , tức là  $X \subseteq X^+$ .

Các tính chất của bao đóng của tập thuộc tính  $X^+$

Nếu  $X, Y$  là các tập con của tập thuộc tính  $Q$  thì ta có các tính chất sau đây:

1. *Tính phản xạ:*  $X \subseteq X^+$
2. *Tính đơn điệu:* Nếu  $X \subseteq Y$  thì  $X^+ \subseteq Y^+$
3. *Tính lũy đẳng:*  $X^{++} = X^+$
4.  $(XY)^+ \supseteq X^+Y^+$
5.  $(X^+Y)^+ = (XY^+)^+ = (X^+Y^+)^+$
6.  $X \rightarrow Y \in F^+ \Leftrightarrow Y \subseteq X^+$
7.  $X \rightarrow Y \Leftrightarrow Y^+ \subseteq X^+$

(Để giáo trình không bị ảnh hưởng quá nặng về lý thuyết toán, chúng tôi không muốn đi sâu về các khái niệm  $F^+$ ,  $X^+$  cũng như việc chứng minh các tính chất của  $F^+$ ,  $X^+$ . Bạn đọc có thể tham khảo thêm ở tài liệu tham khảo [2])

### 5.3.3. Bài Toán Thành Viên

Qua phần trên ta nhận thấy  $X^+$  được định nghĩa thông qua  $F^+$ . Vấn đề nảy sinh khi nghiên cứu lý thuyết CSDL là: Cho trước tập các phụ thuộc hàm  $F$  và một phụ thuộc hàm  $f$ , bài toán kiểm tra có hay không  $f \in F^+$  gọi là *bài toán thành viên*.

Để giải quyết bài toán thành viên thật sự không đơn giản; vì mặc dù  $F$  là rất nhỏ nhưng  $F^+$  thì có thể rất lớn. Tuy nhiên ta có thể giải bằng cách tính  $X^+$  và so sánh  $X^+$  với tập  $Y$ . Dựa vào tính chất  $X \rightarrow Y \in F^+ \Leftrightarrow Y \subseteq X^+$ , ta có ngay câu trả lời  $X \rightarrow Y \in F^+$  hay không? Như vậy thay vì giải bài toán thành viên ta đưa về giải bài toán *tìm bao đóng của tập thuộc tính*.

### 5.3.4. Thuật Toán Tìm Bao Đóng Của Một Tập Thuộc Tính

#### Thuật toán 5.1

*Thuật toán tìm bao đóng với độ phức tạp  $O(N^2)$ , với  $N$  là số lượng thuộc tính của lược đồ quan hệ  $Q$ .*

Dữ Liệu Vào  $Q, F, X \subseteq Q^+$



Dữ Liệu Ra  $X^+$

*Bước 1:* Đặt  $X^+ = X$

*Bước 2:* Temp =  $X^+$

$\forall f \quad U \rightarrow V \in F$   
 if ( $U \subseteq X^+$ )  
 $X^+ = X^+ \cup V.$   
 $F = F - f;$

*Bước 3:* if ( $X^+ = \text{Temp}$ )

“ $X^+$  chính là kết quả cần tìm “

Dừng thuật toán

else

trở lại *Bước 2:*

#### Ví dụ 5.4:

Cho lược đồ quan hệ  $Q(ABCDEFGH)$  và tập phụ thuộc hàm  $F$

$F = \{$   $f_1: B \rightarrow A;$   
 $f_2: DA \rightarrow CE;$   
 $f_3: D \rightarrow H;$   
 $f_4: GH \rightarrow C;$   
 $f_5: AC \rightarrow D\}$

Tìm bao đóng của các tập  $X = \{AC\}$  dựa trên  $F$ .

Giải:

$X^+ = AC$

Do  $f_1, f_2, f_3, f_4$  không thoả.  $f_5$  thoả :

$X^+ = AC\underline{D}$

Lập lại bước 2.  $f_1$  không thoả,  $f_2$  thoả:

$X^+ = ACDE\underline{E}$

$f_3$  thoả :

$X^+ = ACDEH$

Đến đây rõ ràng không có phụ thuộc hàm nào làm thay đổi  $X^+$  nữa, thuật toán dừng lại và kết quả  $X^+ = ACDEH$

Chú ý rằng bạn đọc hãy nắm thật kỹ thuật toán này – nó mở đầu cho một loạt ứng dụng quan trọng về sau. Tiếp theo, chúng tôi nêu lên một thuật toán tìm bao đóng với độ phức tạp tuyến tính để các bạn tham khảo.

### Thuật toán 5.2

*Thuật toán tìm bao đóng với độ phức tạp tuyến tính[3]*

*Bước 1:*

Xây dựng mảng một chiều COUNT

Với COUNT(i) là số thuộc tính về trái của phụ thuộc hàm thứ i

*Bước 2:*

Xây dựng mảng LIST với

$LIST(A) = \{X \rightarrow Y \in F, A \in X\}$

(lưu chỉ số phụ thuộc hàm)

*Bước 3:*

$X^+ = X$

*Bước 4:*

Mọi thuộc tính  $A \in X^+$

Giảm COUNT( $X \rightarrow Y$ ) đi một nếu  $A \in X$

Nếu COUNT( $X \rightarrow Y$ ) = 0 thì  $X^+ = X^+ \cup Y$

Quay lại duyệt thuộc tính kế tiếp trong  $X^+$  cho đến khi nào duyệt hết mọi phần tử của  $X^+$  thì dừng lại. Kết quả  $X^+$  là bao đóng cần tìm.

## 5.4. KHOÁ CỦA LỰỢC ĐỒ QUAN HỆ - MỘT SỐ THUẬT TOÁN TÌM KHOÁ

### 5.4.1. Định Nghĩa Khoá Của Quan Hệ (relation key)

Cho quan hệ  $Q(A_1, A_2, \dots, A_n)$  được xác định bởi tập thuộc tính  $Q^+$  và tập phụ thuộc hàm  $F$  định nghĩa trên  $Q$ , cho  $K \subseteq Q^+$ .

$K$  là một khoá của  $Q$  nếu thoả đồng thời cả hai điều kiện sau:

1.  $K \rightarrow Q^+ \in F^+$  (hay  $K_F^+ = Q^+$ )

( $K$  chỉ thoả điều kiện 1 thì được gọi là siêu khoá)

2. Không tồn tại  $K' \subset K$  sao cho  $K'^+ = Q^+$

Một lược đồ quan hệ có thể có nhiều siêu khoá, nhiều khoá.

#### 5.4.2. Thuật Toán Tìm Một Khoá Của Một Lược Đồ Quan Hệ Q

##### Thuật toán 5.3

$K = Q^+$ ;

While  $A \in K$  do

if  $(K - A)^+ = Q^+$  then  $K = K - A$

K còn lại chính là một khoá cần tìm.

Nếu muốn tìm các khoá khác (nếu có) của lược đồ quan hệ, ta có thể thay đổi thứ tự loại bỏ các phần tử của K.

##### Ví dụ 5.7

Cho lược đồ quan hệ Q(ABC) và tập phụ thuộc hàm

$F = \{ A \rightarrow B;$

$A \rightarrow C;$

$B \rightarrow A \}$

Hãy tìm một khoá của Q.

**Giải:**

$K = \{A, B, C\}$

Loại thuộc tính A, do  $(K - A)^+ = Q^+$  nên  $K = \{B, C\}$

thuộc tính B không loại được do  $(K - B)^+ \neq Q^+$  nên  $K = \{B, C\}$

Loại thuộc tính C, do  $(K - C)^+ = Q^+$  nên  $K = \{B\}$ .

Vậy một khoá của Q là B.

#### 5.4.3. Thuật Toán Tìm Tất Cả Các Khoá Của Một Lược Đồ Quan Hệ

##### Thuật toán 5.4 (thuật toán cơ bản)

*Bước 1:* Xác định tất cả các tập con của Q

Để xác định tất cả các tập con của một lược đồ quan hệ  $Q(A_1, A_2, \dots, A_n)$  ta lần lượt duyệt tất cả  $2^n - 1$  tập hợp con khác rỗng của  $Q^+$  (n là số thuộc tính của lược đồ quan hệ Q), kết quả tìm được giả sử là các tập thuộc tính:  $S = \{X_1, X_2, \dots, X_{2^n-1}\}$

*Bước 2:* Tính  $X_i^+$

**Bước 3:** Nếu  $X_i^+ = Q^+$  thì  $X_i$  là siêu khoá

Nếu một tập con  $X_i$  ( $i = 1.., 2_{n-1}$ ) của  $Q^+$  có bao đóng đúng bằng  $Q^+$  thì tập con đó (theo định nghĩa trên) là một siêu khoá của  $Q$ .

Giả sử sau bước này có  $m$  siêu khoá:  $S = \{S_1, S_2, \dots, S_m\}$

**Bước 4**

*Xây dựng tập chứa tất cả các khoá của  $Q$  từ tập  $S$*

Xét mọi  $S_i, S_j$  con của  $S$  ( $i \neq j$ ), nếu  $S_i \subset S_j$  thì ta loại  $S_j$  ( $i, j=1..m$ ), kết quả còn lại chính là tập tất cả các khoá cần tìm.

### Ví dụ 5.8

Tìm tất cả các khoá của lược đồ quan hệ  $Q$  và tập phụ thuộc hàm  $F$  được cho như sau:

$Q(A, B, C);$

$F = \{A \rightarrow B;$

$A \rightarrow C;$

$B \rightarrow A\}$

$X_i$	$X_i^+$	Siêu khoá	khóa
A	$Q^+$	A	A
B	$Q^+$	B	B
C	C		
AB	$Q^+$	AB	
AC	$Q^+$	AC	
BC	$Q^+$	BC	
ABC	$Q^+$	ABC	

Vậy lược đồ quan hệ  $Q$  có hai khoá là:  $\{A\}$  và  $\{B\}$

Thuật toán trên thì dễ hiểu, dễ cài đặt, tuy nhiên nếu với  $n$  khá lớn thì phép duyệt để tìm ra tập tất cả các tập con của tập  $Q^+$  là điều không hiệu quả, do vậy cần thu hẹp không gian duyệt. Chúng ta sẽ nghiên cứu thuật toán cải tiến theo hướng giảm số thuộc tính của tập cần duyệt.

Chú ý rằng thuật toán này tìm được tất cả các siêu khoá, tất cả các khoá

**Thuật toán 5.5** (thuật toán cải tiến)

Trước khi đi vào thuật toán cải tiến, ta cần đưa thêm một số khái niệm sau:

-Tập nguồn(TN) chứa tất cả các thuộc tính có xuất hiện ở vế trái và không xuất hiện ở vế phải của tập phụ thuộc hàm. Những thuộc tính không tham gia vào bất kỳ một phụ thuộc hàm nào thì cũng đưa vào tập nguồn.

-Tập đích chứa tất cả các thuộc tính có xuất hiện ở vế phải và không xuất hiện ở vế trái của tập phụ thuộc hàm.

-Tập trung gian(TG) chứa tất cả các thuộc tính vừa tham gia vào vế trái vừa tham gia vào vế phải.

Dữ liệu vào:                    Lược đồ quan hệ phổ quát Q và tập phụ thuộc dữ liệu F

Dữ liệu ra:                    Tất cả các khoá của quan hệ

**Bước 0.**                    Tìm tập thuộc tính nguồn(TN), tập thuộc tính trung gian(TG)

    Tìm tất cả các tập con của tập trung gian gọi là  $X_i$  (bằng phương pháp duyệt nhị phân)

    if tập trung gian =  $\emptyset$  then

        Tập Khoá = Tập nguồn ;kết thúc

    Ngược lại

    Qua bước 1

**Bước 1** Tìm tất cả các tập con của tập trung gian:  $X_i$

    :                     $S = \emptyset$

$\forall X_i \in$  tập trung gian

        if             $(\text{Tập nguồn} \cup X_i)^+ = Q^+$  then

$S = S \cup \{ \text{Tập nguồn} \cup X_i \}$

    {S là tập các siêu khoá cần tìm}

**Bước 2:**                    Tính  $TN \cup X_i$

**Bước 3:**                    Tính  $(TN \cup X_i)^+$

**Bước 4:** Nếu  $X_i^+ = Q^+$  thì  $X_i$  là siêu khoá

Nếu một tập con  $TN \cup X_i$  có bao đóng đúng bằng  $Q^+$  thì  $TN \cup X_i$  là một siêu khoá của  $Q$ .

Giả sử sau bước này có  $m$  siêu khoá:  $S = \{S_1, S_2, \dots, S_m\}$

**Bước 5**

*Xây dựng tập chứa tất cả các khoá của  $Q$  từ tập  $S$*

Xét mọi  $S_i, S_j$  con của  $S$  ( $i \neq j$ ), nếu  $S_i \subset S_j$  thì ta loại  $S_j$  ( $i, j=1..m$ ), kết quả còn lại chính là tập tất cả các khoá cần tìm.

**Ví dụ 5.9** (Giải lại bài tập ở ví dụ 5.8)

Áp dụng thuật toán cải tiến ta có lời giải như sau:

$TN = \{\phi\}$  ;  $TG = \{A, B\}$

Gọi  $X_i$  là các tập con của tập  $TG$ :

$X_i$	$(TN \cup X_i)$	$(TN \cup X_i)^+$	Siêu khoá	khóá
$\phi$	$\phi$	$\phi$		
A	A	$Q^+$	A	A
B	B	$Q^+$	B	B
AB	AB	$Q^+$	AB	

Vậy quan hệ trên có hai khoá là : [A] và [B]

**Chú ý :**

Thuật toán cải tiến này tìm được tất cả các khoá, nhưng không chắc tìm ra tất cả các siêu khoá.

## 5.5. PHỦ TỐI THIỂU (minimal cover)

**5.5.1. Tập Phụ Thuộc Hàm Tương Đương (equivalent functional dependancy)**

Cho  $F$  và  $G$  là hai tập phụ thuộc hàm, ta nói  $F$  và  $G$  tương đương (hay  $F$  phủ  $G$  hoặc  $G$  phủ  $F$ ) và ký hiệu là  $F^+ = G^+$  nếu và chỉ nếu mỗi phụ thuộc hàm thuộc  $F$  đều thuộc  $G^+$  và mỗi phụ thuộc hàm thuộc  $G$  đều thuộc  $F^+$ .

Chẳng hạn cho lược đồ quan hệ  $Q(ABCDEFGH)$ , thì hai tập phụ thuộc hàm  $F$  và  $G$  (xác định trên  $Q$ ) là tương đương.

$$F = \{B \rightarrow A; DA \rightarrow CE; D \rightarrow H; GH \rightarrow C; AC \rightarrow D; DG \rightarrow C\}$$

$$G = \{B \rightarrow A; DA \rightarrow CE; D \rightarrow H; GH \rightarrow C; AC \rightarrow D; BC \rightarrow AC;$$

$$BC \rightarrow D; DA \rightarrow AH; AC \rightarrow DEH\}$$

Bạn đọc hãy kiểm chứng lại ví dụ nhận xét này bằng cách sử dụng định nghĩa về tập phụ thuộc hàm tương đương và tính chất  $X \rightarrow Y \in F^+ \Leftrightarrow Y \subseteq X^+$

### Ví dụ 5.5:

Chẳng hạn hai tập phụ thuộc hàm sau là tương đương:

$$Q(A,B,C)$$

$$F = \{A \rightarrow B; A \rightarrow C; B \rightarrow A; C \rightarrow A; B \rightarrow C\}$$

$$G = \{A \rightarrow B; C \rightarrow A; B \rightarrow C\}$$

(việc chứng minh xem như bài tập dành cho bạn đọc)

### 5.5.2. Phủ Tối Thiểu

Để có thể phục vụ quá trình thiết kế cơ sở dữ liệu, cần đưa thêm khái niệm *tập phụ thuộc hàm tối thiểu*.

#### Bổ đề

Mỗi tập các phụ thuộc hàm  $F$  đều được phủ bởi tập các phụ thuộc hàm  $G$  mà vế phải của các phụ thuộc hàm  $G$  chỉ gồm một thuộc tính.

#### Định nghĩa

$F$  được gọi là một tập phụ thuộc hàm tối thiểu nếu  $F$  thỏa đồng thời ba điều kiện sau:

Điều kiện a) Vế phải của  $F$  chỉ có một thuộc tính.

Điều kiện b) Không  $\exists f: X \rightarrow A \in F$  và  $Z \subset X$  mà:

$$F^+ = (F - (X \rightarrow A) \cup (Z \rightarrow A))^+$$

Điều kiện c) Không  $\exists X \rightarrow A \in F$  mà:

$$F^+ = (F - (X \rightarrow A))^+$$

Trong đó vế phải của mỗi phụ thuộc hàm ở điều kiện a) chỉ có một thuộc tính, nên bảo đảm không có thuộc tính nào ở vế phải là dư thừa. điều kiện b) bảo đảm không có một thuộc tính nào tham gia vế trái của phụ thuộc hàm là dư thừa. điều kiện c) bảo đảm cho tập  $F$  không có một phụ thuộc hàm nào là dư thừa.

Chú ý rằng một tập phụ thuộc hàm luôn tìm ra ít nhất một phủ tối thiểu và nếu thứ tự các phụ thuộc hàm trong tập  $F$  là khác nhau thì có thể sẽ thu được những phủ tối thiểu khác nhau.

### 5.5.3. Thuật Toán Tìm Phủ Tối Thiểu

#### Thuật toán 5.6

Dữ liệu vào : Lược đồ quan hệ ban đầu  $Q$  và tập phụ thuộc hàm  $F$ , số lượng

phụ thuộc hàm trong  $F$  là  $m$ .

Dữ liệu ra : Tập phụ thuộc hàm tối thiểu của  $F$

*Bước 1:*

Tách vế phải mỗi phụ thuộc hàm trong  $F$  sao cho vế phải của mỗi phụ thuộc hàm chỉ chứa một thuộc tính (điều này luôn thực hiện được do bổ đề trên)

$$\forall f: X \rightarrow Y \in F$$

$$\forall A \in Y$$

$$g = X \rightarrow A$$

$$F = F \cup g$$

$$m = m + 1$$

Cuối  $\forall$

Cuối  $\forall$

*Bước 2.* Tìm tập phụ thuộc hàm đầy đủ bằng cách loại bỏ các thuộc tính dư thừa ở vế trái của từng phụ thuộc hàm.

$$\forall f X \rightarrow A \in F$$

$$\forall B \in X$$

$$X' = X - B$$

$$\text{If } (X' \rightarrow A \in F^+) \quad X = X'$$

Cuối  $\forall$

Cuối  $\forall$

*Chú ý:*



Việc tìm tất cả các tập  $X' \subseteq X$  theo thuật toán trên hoàn toàn thay thế được việc tìm  $X'$  cách tìm các tập con của  $X$ .

**Bước 3.** Loại bỏ các phụ thuộc hàm dư thừa trong  $F$ .

$\forall f \in F$

$G = F - f$  {loại  $f$  ra khỏi  $F$ . và lưu  $\{ F - f \}$  vào  $G$ }

If  $(F^+ = G^+)$  {gọi thủ tục kiểm tra  $F, G$  tương đương ở dưới}

$F = G$  {cập nhật lại  $F$  mới}

Cuối  $\forall$

### Ví dụ 5.6

Cho lược đồ quan hệ  $Q$  và tập phụ thuộc  $F$  như sau:

$Q(ABCD)$

$F = \{ AB \rightarrow CD;$

$B \rightarrow C;$

$C \rightarrow D \}$

Hãy tìm phủ tối thiểu của  $F$ .

Giải:

kết quả của bước 1 là:

$F = \{ AB \rightarrow C; AB \rightarrow D; B \rightarrow C; C \rightarrow D \}$

kết quả của bước 2 là:

$F = \{ B \rightarrow C; B \rightarrow D; B \rightarrow C; C \rightarrow D \}$

kết quả của bước 3 cho phủ tối thiểu:

$Q(ABCD)$

$F = \{ B \rightarrow C; C \rightarrow D \}$

## 5.6. DẠNG CHUẨN CỦA LƯỢC ĐỒ QUAN HỆ

Khi thiết kế một hệ thống thông tin, thì việc lập lược đồ CSDL đạt đến một tiêu chuẩn nào đó là một việc làm quan trọng. Chất lượng của hệ thống thông tin phụ thuộc rất nhiều vào lược đồ CSDL này. Việc xác định chuẩn cho một lược đồ quan hệ có liên quan mật thiết với thuật toán tìm khoá. Có thể

khẳng định rằng thuật toán tìm khoá là một trong những thuật toán quan trọng của lý thuyết thiết kế cơ sở dữ liệu.

Chất lượng thiết kế của một lược đồ CSDL có thể được đánh giá dựa trên nhiều tiêu chuẩn trong đó sự trùng lặp thông tin và chi phí kiểm tra các ràng buộc toàn vẹn là hai tiêu chuẩn quan trọng. Sau đây là một số dạng chuẩn để đánh giá mức độ tốt/xấu của một lược đồ cơ sở dữ liệu.

Trước hết, chúng ta cùng tìm hiểu một số khái niệm liên quan.

### 5.6.1. Một Số Khái Niệm Liên Quan Đến Các Dạng Chuẩn

*Thuộc tính khoá/không khoá*

A là một thuộc tính khoá nếu A có tham gia vào bất kỳ một khoá nào của quan hệ, ngược lại A gọi là thuộc tính không khoá.

#### Ví dụ 5.10

**Cho lược đồ quan hệ Q(ABC) và tập phụ thuộc hàm**

$F = \{ A \rightarrow B;$

$A \rightarrow C;$

$B \rightarrow A \}$

*Có hai khóa là A và B. khi đó thuộc tính khoá là A, B; thuộc tính không khoá là: C.*

*Thuộc tính phụ thuộc đầy đủ- phụ thuộc hàm đầy đủ.*

A là một thuộc tính phụ thuộc đầy đủ vào tập thuộc tính X nếu  $X \rightarrow A$  là một phụ thuộc hàm đầy đủ (tức là không tồn tại  $X' \subset X$  sao cho  $X' \rightarrow A \in F^+$ )

#### Ví dụ 5.10

**Cho lược đồ quan hệ Q(ABC) và tập phụ thuộc hàm**

$F = \{ A \rightarrow B$

$A \rightarrow C;$

$AB \rightarrow C$

$\}$

thì  $A \rightarrow B$ ;  $A \rightarrow C$  là các phụ thuộc hàm đầy đủ. Phụ thuộc hàm  $AB \rightarrow C$  không là phụ thuộc hàm đầy đủ vì có  $A \rightarrow C$ .

Chú ý rằng, một phụ thuộc hàm mà về trái chỉ có một thuộc tính là phụ thuộc hàm đầy đủ.

### 5.6.2. Dạng Chuẩn Một (First Normal Form)

Lược đồ quan hệ Q được gọi là đạt dạng chuẩn 1 (1NF) nếu và chỉ nếu toàn bộ các thuộc tính của Q đều mang giá trị đơn.

Chẳng hạn xét *quan hệ*

MASV	HOTEN	MONHOC	DIEMTHI
00CDT H189	Nguyễn Văn Thành	<i>Kỹ Thuật Lập Trình</i>	8
		<i>Cơ Sở Dữ Liệu</i>	9
		<i>Cấu Trúc Dữ Liệu</i>	7
00CDT H211	Trần Thu Hà	<i>Kỹ Thuật Lập Trình</i>	5

Lược đồ quan hệ này không đạt dạng chuẩn 1 vì các thuộc tính MONHOC, DIEMTHI không mang giá trị đơn (chẳng hạn sinh viên Nguyễn Văn Thành có thuộc tính môn học là *Kỹ Thuật Lập Trình*, *Cơ Sở Dữ Liệu*, *Cấu Trúc Dữ Liệu*).

Ta hoàn toàn có thể đưa quan hệ trên về dạng chuẩn 1 như sau:

MASV	HOTEN	MONHOC	DIEMTHI
00CD TH189	Nguyễn Văn Thành	<i>Kỹ Thuật Lập Trình</i>	9
00CD TH189	Nguyễn Văn Thành	<i>Cơ Sở Dữ Liệu</i>	8
00CD TH189	Nguyễn Văn Thành	<i>Cấu Trúc Dữ Liệu</i>	7
00CD TH211	Trần Thu Hà	<i>Kỹ Thuật Lập Trình</i>	5

Chú ý rằng nếu ta không nói gì thêm, thì lược đồ quan hệ đang xét ít nhất là đạt dạng chuẩn 1.

### 5.6.3. Dạng Chuẩn 2 (second normal form)

Một lược đồ quan hệ Q đạt dạng chuẩn 2 nếu Q đạt dạng chuẩn 1 và tất cả các thuộc tính không khoá của Q đều phụ thuộc đầy đủ vào khoá.

Nếu một lược đồ quan hệ không đạt chuẩn 2 thì ta nói nó đạt dạng chuẩn 1.

Chẳng hạn xét lược đồ quan hệ

$Q(A,B,C,D)$  và

$F = \{ AB \rightarrow C, D;$

$B \rightarrow D;$

$C \rightarrow A \}$

Khoá là  $\{A,B\}$  và  $\{B,C\}$ . Do đó  $D$  là thuộc tính không khoá,  $A, B \rightarrow D$  không là phụ thuộc hàm đầy đủ vì có  $B \rightarrow D$ .

Vậy  $Q$  đạt chuẩn 1.

**Ví dụ 5.11:**

Xác định dạng chuẩn của lược đồ quan hệ sau.

$Q(G,M,V,N,H,P)$

$F = \{ G \rightarrow N; G \rightarrow H; G \rightarrow P; M \rightarrow V; NHP \rightarrow M \}$

Để thấy khoá của  $Q$  là  $G$ .

Thuộc tính không khoá là  $M, V, N, H, P$ .

Do các phụ thuộc hàm  $G \rightarrow M; G \rightarrow V; G \rightarrow N; G \rightarrow H; G \rightarrow P$  là các phụ thuộc hàm đầy đủ, nên lược đồ quan hệ  $Q$  đạt dạng chuẩn 2

Hệ quả:

- $Q$  đạt 2NF nếu  $Q$  là 1NF và tập thuộc tính không khoá của  $Q$  bằng rỗng.

-Nếu khoá của quan hệ có một thuộc tính thì quan hệ đó ít nhất đạt chuẩn 2.

**Ví dụ 5.12:**

$Q(ABCDEH)$

$F = \{ A \rightarrow E; C \rightarrow D; E \rightarrow DH \}$

Để thấy khoá của  $Q$  là  $K = \{ABC\}$

$D$  là thuộc tính không khoá. và  $C \rightarrow D$ , vì  $C$  là tập con thực sự của khoá nên  $Q$  không đạt dạng chuẩn 2.

**Dạng Chuẩn 3 (third normal form)**

Một lược đồ quan hệ  $Q$  đạt dạng chuẩn 3 nếu mọi phụ thuộc hàm  $X \rightarrow A \in F^+$  ( $F$  là tập phụ thuộc không hiển nhiên định nghĩa trên  $Q$ ,  $A$  là thuộc tính đơn,  $X$  là tập thuộc tính con của tập  $Q^+$ ), thì một trong hai điều kiện sau được thoả:

Hoặc  $X$  là một siêu khoá của  $Q$

Hoặc  $A$  là một thuộc tính khoá

*Nhận xét:*

Nếu  $Q$  đạt chuẩn 3 thì  $Q$  đạt chuẩn 2

### **Ví dụ 5.13**

Cho lược đồ quan hệ  $Q(ABCD)$

$F = [AB \rightarrow C ; D \rightarrow B \ C \rightarrow ABD]$

$K1 = [AB]; K2 = [AD]; K3 = [C]$

là các khoá, vậy  $Q$  không có thuộc tính không khoá nên  $Q$  đạt chuẩn 3

Hệ quả

Nếu lược đồ quan hệ  $Q, F$  mà  $Q$  không có thuộc tính không khoá thì  $Q$  đạt chuẩn 3.

### **Ví dụ 5.14**

Xác định dạng chuẩn của lược đồ quan hệ sau.

$Q(NGPM)$

$F = \{NGP \rightarrow M;$

$M \rightarrow P\}$

Để thấy các khoá của  $Q$  là  $\{NGP\}, \{NGM\}$

$NGP \rightarrow M$  có về trái là siêu khoá

$M \rightarrow P$  có về phải là thuộc tính khoá.

Nên  $Q$  đạt chuẩn 3.

### **5.6.5. Dạng Chuẩn BC (Boyce Codd normal form)**

Một lược đồ quan hệ  $Q$  ở dạng chuẩn BC nếu với mỗi phụ thuộc hàm không hiển nhiên  $X \rightarrow A \in F$  thì  $X$  là một siêu khoá của  $Q$ .

*Nhận xét:* Nếu  $Q$  đạt chuẩn BC thì  $Q$  đạt chuẩn 3

**Ví dụ 5.15**

Xác định dạng chuẩn của lược đồ quan hệ sau.

$Q(ACDEIB)$

$F = \{ACD \rightarrow E; CE \rightarrow AD\}$

Để thấy  $Q$  có hai khoá là:  $ACD$  và  $CE$ . Các phụ thuộc hàm của  $F$  đều có vế trái là siêu khoá, nên  $Q$  đạt dạng chuẩn BC.

**ĐỊNH LÝ :** Các lớp dạng chuẩn của một lược đồ quan hệ có quan hệ lồng nhau: nghĩa là lớp sau nằm trọn trong lớp trước.  $BCNF \subset 3NF \subset 2NF \subset 1NF$

**Ví dụ 5.16**

Chẳng hạn cho lược đồ quan hệ  $Q(ABCD)$  và  $F = [AB \rightarrow C; D \rightarrow B; C \rightarrow ABD]$  thì  $Q$  đạt chuẩn 3NF nhưng không là BCNF

Nếu  $F = [B \rightarrow D, A \rightarrow C, C \rightarrow ABD]$  thì  $Q$  đạt dạng chuẩn 2NF nhưng không là 3 NF.

**Dạng chuẩn của một lược đồ cơ sở dữ liệu** là dạng chuẩn thấp nhất của các lược đồ quan hệ con.

**Chú ý:** Các dạng chuẩn cao hơn như dạng chuẩn bốn (với phụ thuộc đa trị), dạng chuẩn năm (với phụ thuộc chiếu kết) có thể xem các tài liệu tham khảo đã chỉ ra.

**BÀI TẬP**

5.1. a) Cho lược đồ quan hệ  $Q(ABCD)$ ,  $r$  là một quan hệ trên  $Q$ .

$r$			
A	B	C	D
a1	b1	c1	d1
a1	b2	c1	d1
a1	b3	c2	d1
a2	b2	c2	d2

phụ thuộc hàm nào sau đây không thoả  $r$

- a)  $D \rightarrow A$ ;
- b)  $A, C \rightarrow D$ ;
- c)  $CD \rightarrow A$ ;
- d)  $D \rightarrow B$ ;

b. Cho lược đồ quan hệ  $Q(ABCD)$ ,  $r$  là quan trên  $Q$  được cho như sau:

A	B	C	D
a1	b1	c1	d2
a3	b1	c2	d1
a1	b1	c2	d2

Những phụ thuộc hàm nào sau đây thoả  $r$  ?

$AB \rightarrow D$ ;       $C \rightarrow B$ ;       $B \rightarrow C$ ;  $BC \rightarrow A$ ;       $BD \rightarrow A$ .

c. Cho lược đồ quan hệ  $Q(ABCD)$ ,  $r$  là quan hệ được cho như sau:

A	B	C	D
x	u	x	y
y	x	z	x
z	y	y	y
y	z	w	z

Những phụ thuộc hàm nào sau đây không thoả  $r$  ?

$A \rightarrow B; \quad A \rightarrow C; \quad B \rightarrow A; \quad C \rightarrow D; \quad D \rightarrow C; \quad D \rightarrow A$

**5.2.** a. Cho lược đồ quan hệ  $Q(ABCD)$  và tập phụ thuộc hàm

$F = \{ \quad A \rightarrow B;$   
 $\quad \quad BC \rightarrow D \}$ .

Những phụ thuộc hàm nào sau đây thuộc  $F^+$  ?

$C \rightarrow D; \quad A \rightarrow D; \quad AD \rightarrow C; \quad AC \rightarrow D; \quad BC \rightarrow A; \quad B \rightarrow CD.$

b. Cho lược đồ quan hệ  $Q(ABCDEFGH)$  và tập phụ thuộc hàm

$F = \{ \quad AB \rightarrow C;$   
 $\quad \quad B \rightarrow D;$   
 $\quad \quad CD \rightarrow E;$   
 $\quad \quad CE \rightarrow GH;$   
 $\quad \quad G \rightarrow A \}$

Những phụ thuộc hàm nào sau đây không thuộc vào  $F^+$  ?

$AB \rightarrow E; \quad AB \rightarrow GH; \quad CGH \rightarrow E; \quad CB \rightarrow E; \quad GB \rightarrow E.$

c) Cho lược đồ quan hệ  $Q, F$  như sau:

với  $Q=(ABCD)$

$F = [ \quad A \rightarrow B;$   
 $\quad \quad A \rightarrow C ]$ .

Trong các phụ thuộc hàm sau, những phụ thuộc hàm được suy ra từ  $F$  ?

$A \rightarrow D; \quad C \rightarrow D; \quad AB \rightarrow B; \quad \quad BC \rightarrow A; \quad \quad A \rightarrow BC$

**5.3.** Cho lược đồ quan hệ  $Q(ABCD)$  và tập phụ thuộc hàm

$F = \{ \quad A \rightarrow D;$   
 $\quad \quad D \rightarrow A;$   
 $\quad \quad AB \rightarrow C \}$

a. Tính  $AC^+$

b. Chứng minh  $BD \rightarrow C \in F^+$

**5.4.** a)  $Q(ABCDEG)$

Cho  $F = \{ AB \rightarrow C;$



$C \rightarrow A;$   
 $BC \rightarrow D;$   
 $ACD \rightarrow B;$   
 $D \rightarrow EG;$   
 $BE \rightarrow C ;$   
 $CG \rightarrow BD;$   
 $CE \rightarrow AG\}$

$X=[BD], X^+=?$

$Y=[CG], Y^+=?$

**b)** Cho lược đồ quan hệ Q và tập phụ thuộc hàm F

$F=\{$      $AB \rightarrow E;$   
            $AG \rightarrow I;$   
            $BE \rightarrow I;$   
            $E \rightarrow G ;$   
            $GI \rightarrow H \}$

Chứng minh rằng  $AB \rightarrow GH$ .

c. Tương tự cho tập phụ thuộc hàm

$F = \{$      $AB \rightarrow C;$   
            $B \rightarrow D;$   
            $CD \rightarrow E;$   
            $CE \rightarrow GH;$   
            $G \rightarrow A\}$

Chứng minh rằng  $AB \rightarrow E; AB \rightarrow G$

d.  $Q(ABCDEFGHI)$

$F = \{B \rightarrow A; DA \rightarrow CE; D \rightarrow H; GH \rightarrow C; AC \rightarrow D \}$

Hãy tìm một khoá của Q ?

**5.5.** Hãy tìm tất cả các khoá cho lược đồ quan hệ sau:

$Q(\text{BROKER,OFFICE,STOCK,QUANTITY,INVESTOR,DIVIDENT})$

$F=\{$

$\text{STOCK} \rightarrow \text{DIVIDENT}$

INVESTOR → BROKER

INVESTOR, STOCK → QUANTITY

BROKER → OFFICE }

5.6.Q(A,B,C,D)

F=[ AB → C;

D → B;

C → ABD]

Hãy tìm tất cả các khoá của Q

5.7. Cho lược đồ quan hệ Q(MSCD,MSSV,CD,HG) và tập phụ thuộc F như sau:

F=[ MSCD→CD;

CD→MSCD;

CD,MSSV→HG;

MSCD,HG→MSSV;

CD,HG→MSSV;

MSCD,MSSV→HG}

Hãy tìm phủ tối thiểu của F.

5.8 Xác định phủ tối thiểu của tập phụ thuộc hàm sau:

Q(ABCDEG)

F = {AB → C;

C → A;

BC → D;

ACD → B;

D → EG;

BE → C;

CG → BD;

CE → AG}

5.9. Các nhận xét sau đúng (Đ) hay sai (S) ? (kẻ bảng sau và ghi Đ hoặc S cho mỗi câu trên)

a	b	c	d	e	f	g	H

a.Cho Q và F={AB → C; A → B} thì Q đạt dạng chuẩn 1.

- b. Một lược đồ quan hệ Q luôn tìm được ít nhất một khoá.
- c. Nếu  $XY \rightarrow Z$  thì  $X \rightarrow Z$  và  $Y \rightarrow Z$ .
- d. Các thuộc tính không tham gia vào vế phải của bất kỳ phụ thuộc hàm nào thì phải là thuộc tính tham gia vào khoá.
- e. Nếu  $X \rightarrow Y$  và  $YZ \rightarrow W$  thì  $XZ \rightarrow W$
- f. Nếu Q đạt dạng chuẩn một và khoá của Q chỉ có một thuộc tính thì Q đạt dạng chuẩn ba.
- g. Một tập phụ thuộc hàm F có thể có nhiều tập phủ tối thiểu.
- h. Nếu  $X \rightarrow Y$  và  $U \rightarrow V$  thì  $XU \rightarrow YV$ .

**5.10** a. Cho Q(ABCD) và

$$F = \{AB \rightarrow C; D \rightarrow B; C \rightarrow ABD\}.$$

Hãy kiểm tra xem  $AB \rightarrow D$  có thuộc  $F^+$  hay không ?

Hãy tìm tất cả các khoá của lược đồ quan hệ Q. Xác định dạng chuẩn của Q.

b. Cho Q(A,B,C,D)

$$\text{và } F = \{C \rightarrow A; A \rightarrow C; AD \rightarrow B; BC \rightarrow D; AB \rightarrow D; CD \rightarrow B\}$$

Hãy tìm phủ tối thiểu của F.

**5.15.** Cho biết dạng chuẩn của các lược đồ quan hệ sau:

a. Q(ABCDEG);

$$F = [A \rightarrow BC, C \rightarrow DE, E \rightarrow G]$$

b. Q(ABCDEFGH);

$$F = [C \rightarrow AB, D \rightarrow E, B \rightarrow G]$$

c. Q(ABCDEFGH);

$$F = [A \rightarrow BC, D \rightarrow E, H \rightarrow G]$$

d. Q(ABCDEG);

$$F = [AB \rightarrow C; C \rightarrow B; ABD \rightarrow E; G \rightarrow A]$$

e. Q(ABCDEFGHI);

$$F = [AC \rightarrow B; BI \rightarrow ACD; ABC \rightarrow D; H \rightarrow I; ACE \rightarrow BCG, CG \rightarrow AE]$$

















## PHỤ LỤC(MỘT SỐ ĐỀ KIỂM TRA, ĐỀ THI MÔN CSDL)

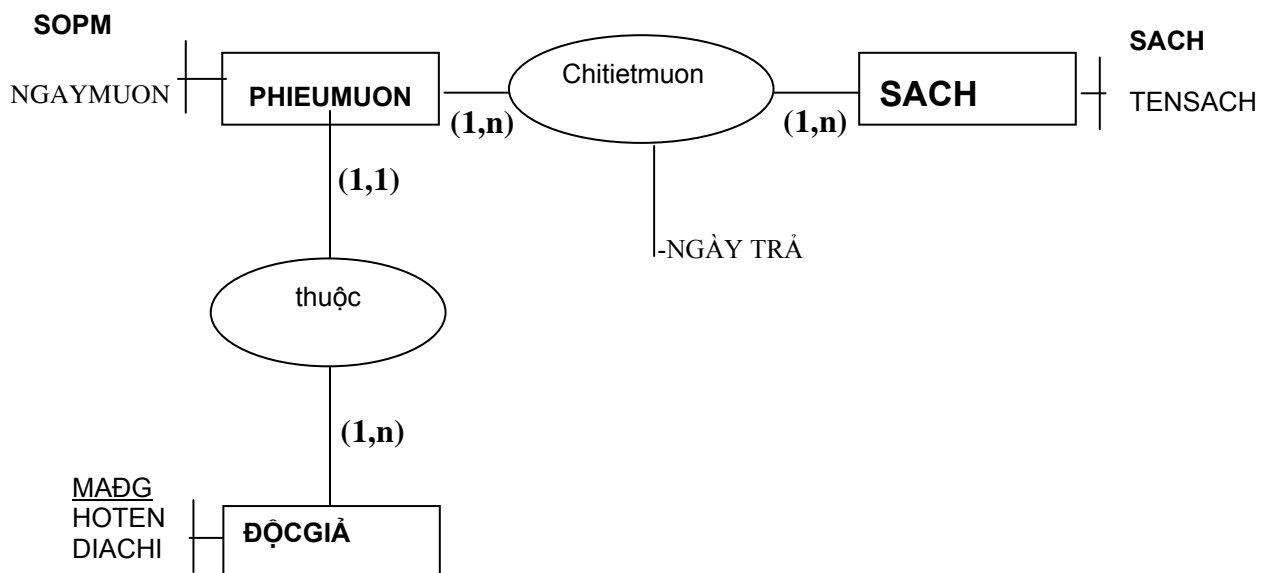
TRƯỜNG CĐ KỸ THUẬT CAO THẮNG

### BÀI KIỂM TRA HỆ SỐ 2 - MÔN CƠ SỞ DỮ LIỆU (bài thứ nhất)

(Thời gian 45 phút)

#### CÂU 1:(3 điểm)

Chuyển mô hình thực thể kết hợp sau sang mô hình dữ liệu quan hệ. Tìm khóa cho mỗi lược đồ quan hệ con.



#### CÂU 2:(3 điểm)

Cho lược đồ cơ sở dữ liệu với 3 lược đồ quan hệ sau:

**Nhanvien(MANV,HOTEN,NGAYSINH,ĐIACHI, MAPB,MACV)**

Mỗi nhân viên có một mã nhân viên (MANV) duy nhất, mỗi MANV xác định các thông tin: họ và tên (HOTEN), ngày sinh (NGAYSINH) - NGÀY SINH là dạng ngày tháng năm, địa chỉ (ĐIACHI), mã phòng ban(MAPB) và mã chức vụ (MACV).

**Phongban (MAPB,TENPB)**

Mỗi phòng ban có một mã phòng ban (MAPB) duy nhất, mỗi MAPB xác định tên phòng ban (TENPB).

**Chucvu(MACV,TENCV,PHUCAP)**

Mỗi chức vụ có một mã chức vụ (MACV) duy nhất, mỗi MACV xác định tên chức vụ (TENCV), phụ cấp chức vụ(PHUCAP)

Câu 1.Tìm khóa cho mỗi lược đồ quan hệ trên.

Câu 2:Dựa vào lược đồ cơ sở dữ liệu trên, hãy thực hiện các yêu cầu sau bằng ngôn ngữ đại số quan hệ

a.Lập danh sách các nhân viên có MAPB là "KT". Danh sách cần MANV, HOTEN,NGAYSINH,ĐIACHI.

b.Lập danh sách các nhân viên có phụ cấp chức vụ. Danh sách cần các thông tin: MANV, HOTEN, ĐIACHI, TENPB, TENCV.

**CÂU 3:(4 điểm)**

a.(1,0 điểm)

Cho lược đồ quan hệ Q(ABCD), r và s là hai quan hệ được cho như sau:

r				s			
A	B	C	D	A	B	C	D
1	0	0	0	2	1	1	1
1	1	0	0	2	2	1	1
1	1	1	0	1	1	1	0
·	·	·	·	x	y	z	v

Tìm  $r - s$

Tính  $r + s$

Tính  $r - s$

b.(1,0 điểm)

Cho hai lược đồ quan hệ Q1(ABC) và Q2(DEF), r và s là hai quan hệ được cho như sau:

r			s		
A	B	C	D	E	F
1	2	3	1	e	f
a	b	c	a	e	f
x	y	z	5	6	7

$$A = D$$

Tìm  $r \succ\prec s$

c.(1,0 điểm)

Cho hai lược đồ quan hệ Q1(ABC) và Q2(DE), r và s là hai quan hệ được cho như sau

r			s	
A	B	C	D	E
1	2	3	3	1
4	5	6	6	2
7	8	9		

$$B > D$$

Tìm  $r \succ\prec s$

d.(1,0 điểm)

Cho quan hệ về khả năng lái các loại máy bay của các phi công:

KHANANG(SỐ HIỆU PHI CÔNG, SỐ HIỆU MÁY BAY)

SỐ HIỆU PHI CÔNG      SỐ HIỆU MÁY BAY

32	102
30	101
30	103
32	103
33	100
30	102
31	102
30	100
31	100

Hãy cho biết các phi công có khả năng lái được cả 3 loại máy bay:

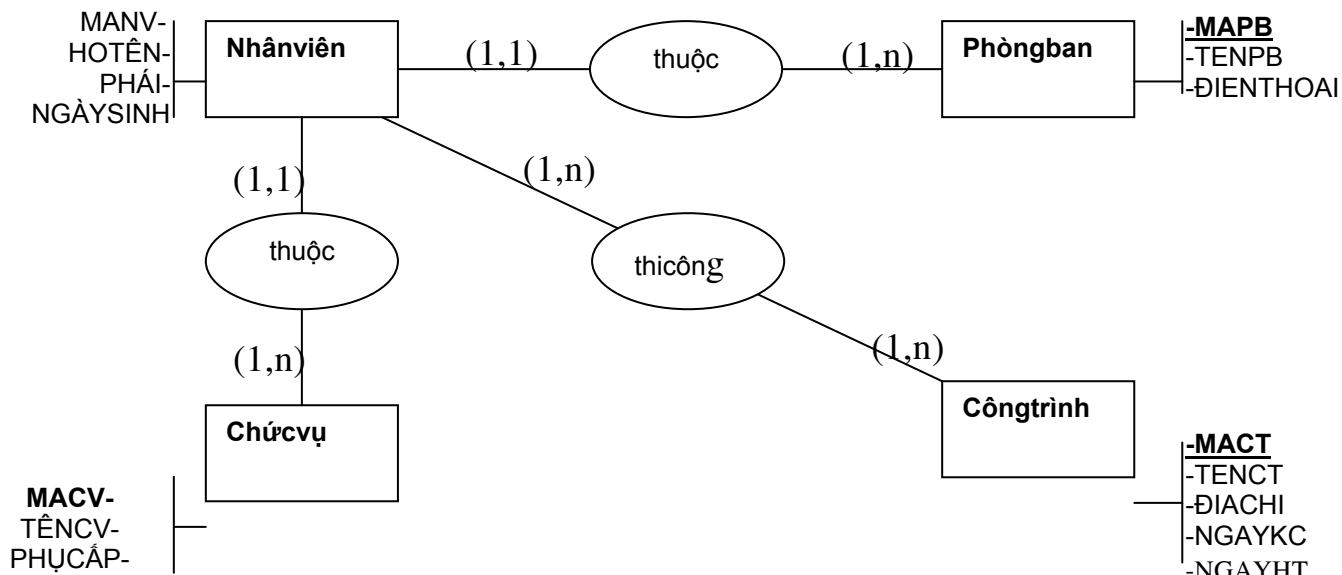
100,101,103 ? Đây là kết quả của phép toán quan hệ nào ?

Hết

**TRƯỜNG CÔNG NGHỆ THÔNG TIN**

**ĐỀ KIỂM TRA GIỮA KỲ**  
MÔN CƠ SỞ DỮ LIỆU  
thời gian làm bài 60 phút

**CÂU I: (2,0đ)**



**CÂU II: (3,0đ)**

1. Cho hai lược đồ quan hệ  $Q_1(ABC)$  và  $Q_2(DEF)$ ,  $r$  và  $s$  là hai quan hệ được cho như

sau:

	r				s		
	A	B	C	D	E	F	
	2	5	7	2	4	3	
	4	2	6	1	5	7	
	1	5	9	5	2	3	

A=D B > F

a. Tính  $r \bowtie s$

b. Tính  $r \bowtie s$

2. Cho hai lược đồ quan hệ  $Q_1(ABCD)$  và  $Q_2(CD)$ ,  $r$  và  $s$  là hai quan hệ được cho như

sau:

	r					s	
	A	B	C	D	C	D	
	3	4	5	6	3	4	
	1	2	3	4	5	6	
	1	2	5	6			

2      3      5      6

1      2      4      5

Tính  $r \div s$

### CÂU III (5,0đ)

Một phòng giáo dục quận muốn lập một hệ thống thông tin để quản lý việc làm thi tốt nghiệp phổ thông cơ sở (lớp 9). Công việc làm thi được tổ chức như sau:

Lãnh đạo phòng giáo dục thành lập nhiều hội đồng thi (mỗi hội đồng thi gồm một trường hoặc một số trường gần nhau). Mỗi hội đồng thi có một mã số duy nhất (MAHĐT), một hội đồng thi xác định tên hội đồng thi(TENHĐT), họ tên chủ tịch hội đồng(TENCT), địa chỉ (ĐCHĐT),điện thoại(ĐTHĐT).

Mỗi hội đồng thi được bố trí cho một số phòng thi, mỗi phòng thi có một số hiệu phòng(SOPT) duy nhất, một phòng thi xác định địa chỉ phòng thi (ĐCPT). Số hiệu phòng thi được đánh số khác nhau ở tất cả các hội đồng thi.

Giáo viên của các trường trực thuộc phòng được điều động đến các hội đồng để coi thi, mỗi trường có thể có hoặc không có thí sinh dự thi, mỗi trường có một mã trường duy nhất (MATR), mỗi mã trường xác định tên trường(TENTR),địa chỉ (ĐCTR), loại hình đào tạo (LHĐT) (Công lập, chuyên, bán công, dân lập,...). Các giáo viên của một trường có thể làm việc tại nhiều hội đồng thi. Một giáo viên có một mã giáo viên(MAGV), một mã giáo viên xác định tên giáo viên (TENGV), chuyên môn giảng dạy (CHUYENMON), chức danh trong hội đồng thi(CHUCDANH).

Các thí sinh dự thi có một số báo danh duy nhất(SOBD), mỗi số báo danh xác định tên thí sinh(TENTS), ngày sinh (NGSINH), giới tính (PHAI), mỗi thí sinh được xếp thi tại một phòng thi nhất định cho tất cả các môn, mỗi thí sinh có thể có chứng chỉ nghề (CCNGHE) hoặc không (thuộc tính CCNGHE kiểu chuỗi, CCNGHE="x" nếu thí sinh có chứng chỉ nghề và CCNGHE bằng rỗng nếu thí sinh không có chứng chỉ nghề).Thí sinh của cùng một trường chỉ dự thi tại một hội đồng thi.

Mỗi môn thi có một mã môn thi duy nhất (MAMT), mỗi mã môn thi xác định tên môn thi(TENMT), buổi thi (BUOI), ngày thi (NGAY). Giả sử toàn bộ các thí sinh trong hội đồng thi đó đều thi chung một số môn do sở giáo dục quy định (có thể thay đổi tùy theo năm). Mỗi môn thi

được tổ chức trong một buổi của một ngày nào đó. Ứng với mỗi môn thi một thí sinh có một điểm thi duy nhất(ĐIEMTHI)

Dựa vào phân tích ở trên, giả sử ta có lược đồ cơ sở dữ liệu sau:

HĐ(MAHĐT, TENHĐT, TENCT, ĐCHĐT, ĐTHĐT)

PT(SOPT, ĐCPT, MAHĐT)

TS(SOBD, TENTS, NGSINH, PHAI, CCNGHE, MATR, SOPT)

MT(MAMT, TENMT, BUOI, NGÀY)

GV(MAGV, TENGV, CHUYENMON, CHUCDANH, MAHĐT, MATR)

TR(MATR, TENTR, ĐCTR, LHĐT)

KQ(SOBD, MAMT, ĐIEMTHI)

#### YÊU CẦU

1. Hãy xác định khóa cho mỗi lược đồ quan hệ trên.

2. Dựa vào lược đồ cơ sở dữ liệu trên, hãy thực hiện các yêu cầu sau bằng SQL.

a. Danh sách các thí sinh thi tại phòng thi có số hiệu phòng thi (SOPT) là "100"

Yêu cầu các thông tin: SOBD, TENTS, NGSINH, TENTR

b. Kết quả của môn thi có mã môn thi (MAMT) là "T" của tất cả các thí sinh có mã trường (MATR) là "NTMK", kết quả được sắp theo chiều giảm dần của điểm thi (ĐIEMTHI).

Yêu cầu các thông tin: SOBD, TENTS, ĐIEMTHI

c. Tổng số thí sinh có chứng chỉ nghề(CCNGHE) của mỗi trường, thông tin cần được sắp theo chiều tăng dần của TENTR.

Yêu cầu các thông tin: MATR, TENTR, SOLUONGCC, trong đó SOLUONGCC là thuộc tính tự đặt.

Hết

(Sinh viên không được sử dụng tài liệu

Cán bộ coi thi không giải thích)



**TRƯỜNG CÔNG NGHỆ THUẬT CAO THẮNG****ĐỀ KIỂM CHƯƠNG 3**  
**MÔN CƠ SỞ DỮ LIỆU**  
thời gian làm bài 90 phútHỌ VÀ TÊN SV:  
MASV:

ĐIỂM

<i>U</i>	<i>CÂ</i>	<i>KẾT QUẢ</i>	<i>CÂU</i>	<i>KẾT QUẢ</i>
1			26	
2			27	
3			28	
4			29	
5			30	
6			31	
7			32	
8			33	
9			34	
10			35	
11			36	
12			37	
13			38	
14			39	
15			40	
16			41	
17			42	
18			43	
19			44	
20			45	
21			46	
22			47	
23			48	
24			49	
25			50	

1. Consider a relation  $Q(A,B,C,D)$  with the following functional dependencies:

$A \rightarrow B$  and  $B \rightarrow C$

The number of superkeys of  $Q$  is:

- (a) 2
  - (b) 3
  - (c) 4
  - (d) 5
  - (e) None of the above
2. Which of the following functional dependencies must be FALSE ?
- a)  $X \rightarrow Y \Rightarrow XZ \rightarrow YZ$
  - b)  $X \rightarrow Y; YW \rightarrow Z \Rightarrow XW \rightarrow Z.$
  - c)  $X \rightarrow Y, X \rightarrow Z \Rightarrow X \rightarrow YZ$
  - d)  $X \rightarrow Y; Z \subseteq Y \Rightarrow X \rightarrow Z$
  - e) None of the above
3. Consider relation  $S(B, G, U, N, A)$  with the FD's:  $BG \rightarrow U, G \rightarrow N, NA \rightarrow B$
- What are all the keys of  $S$  ?
- a)  $\{B, G\}$  and  $\{N, A\}$
  - b)  $\{U, A\}$
  - c)  $\{G, A\}$
  - d)  $\{B, N\}$
  - e)  $\{B, G, U, N\}$
  - f)  $\{B, G, A\}$  and  $\{G, N, A\}$
4. Which one of the following is correct?
- (a) All FDs must involve the attributes of the same relation.
  - (b) All FDs must involve only a single attribute on the left side.
  - (c) All FDs must involve only a single attribute on the right side.
  - (d) All FDs must involve only single attributes on both sides.
  - (e) None of the above

5. Assume the relation R(A, B, C, D, E) is in at least 3NF. Which of the following functional dependencies must be FALSE?
- a. A, B → C
  - b. A, B → D
  - c. C, D → E
  - d. None of the above
6. SQL provides a number of special aggregate functions. Which one of the following is not included in SQL?
- (a) SUM
  - (b) MAX
  - (c) MIN
  - (d) COUNT
  - (e) MEDIAN
7. Which of the following finds all groups meeting stated conditions?
- (a) Select
  - (b) Where
  - (c) Find
  - (d) Having
8. The \_\_\_\_\_ clause is used to restrict the groups returned by a query.
- (a) FROM
  - (b) WHERE
  - (c) HAVING
  - (d) GROUP BY
9. To get all the customers from Hawaii sorted together, which of the following would be used?
- (a) Order By
  - (b) Group By
  - (c) Having
  - (d) Sort

10. The \_\_\_\_ set operator will show all rows common to both tables A and B.
- (a) intersection
  - (b) union
  - (c) difference
  - (d) product
11. Using the product operator, if table A has 2 rows and table B has 4 rows, the number of rows in the product of these two tables is:
- (a) 4.
  - (b) 8.
  - (c) 16.
  - (d) 20.
12. Suppose relation R(A,B,C,D,E) has the following functional dependencies:

$A \rightarrow B$

$B \rightarrow C$

$BC \rightarrow A$

$A \rightarrow D$

$D \rightarrow E$

Which of the following is not a key?

- (a) A
  - (b) E
  - (c) B
  - (d) D
  - (e) (b) and (d)
13. Consider a relation Q with five attributes L V O B Y. You are given the following dependencies:  $L \rightarrow V$ ;  $VO \rightarrow Y$  and  $YB \rightarrow L$ .
- The number of superkeys of Q is: (not superkeys)
- (a).2
  - (b).3
  - (c).4
  - (d).5

14. Use the following tables and data, All of the fields are Integers. The table names are Q

A	B	C	D
1	2	3	4
1	3	5	7
2	3	4	5
2	4	6	8
5	6	7	8

How many records does the following SQL example return?

```
SELECT * FROM Q WHERE A>=5 OR D>=7;
```

- a) 1
- b) 2
- c) 3
- d) 4

15. How many records does the following SQL example return?

```
SELECT sum(A) AS S FROM Q GROUP BY A;
```

a)	b)	c)	d)
S	S	S	S
11	1	4	None of the above
	1	7	
	2	6	
	2	3	
	5	5	

16. The SQL below will return one value. What is it?

```
SELECT sum(A) AS S FROM Q GROUP BY A HAVING count(A)>1;
```

a)	b)	c)	d)
S	S	S	S
2	9	2	None of the above
4		2	

5

17. How many records does the following SQL example return?

```
SELECT count(*) FROM Q GROUP BY A, B;
```

- a) 2
- b) 3
- c) 4
- d) 6
- e) None of the above

18. Use the following tables and data..All of the fields are Integers.The table names are R and

S

R

A	B
6	7
4	2

S

C	D	E
8	4	2
3	5	7
6	2	9

How many records does the following SQL example return?

```
SELECT COUNT(*) FROM R, S WHERE R.A=S.D OR R.B=S.E;
```

- a) 1
- b) 2
- c) 3
- d) 4

19. How many records does the following SQL example return?

```
SELECT COUNT(*) FROM R, S WHERE R.A=S.D;
```

- a) 0
- b) 1

- c) 2  
d) 4
20. Which of the following statements contains an error?
- (a) `SELECT * FROM emp WHERE empid = 493945;`
  - (b) `SELECT empid FROM emp WHERE empid= 493945;`
  - (c) `SELECT empid FROM emp;`
  - (d) `SELECT empid WHERE empid = 56949 AND lastname = 'SMITH';`
21. Which of the following statements will return the names of the products with Product ID 10, 11, or 42?
- (a) `SELECT ProductName FROM products WHERE ProductID IN (10,11,42)`
  - (b) `SELECT ProductName FROM products WHERE ProductID IN 10 OR 11 OR 42`
  - (c) `SELECT ProductName FROM products WHERE ProductID = (10,11,42)`
  - (d) `SELECT ProductName FROM products WHERE ProductID IS (10,11,42)`
  - (e) None of the above
22. Which of the following commands will return the list of product names sorted in ascending alphabetic order?
- (a) `SELECT ProductName FROM products ORDER BY ProductName DESC`
  - (b) `SELECT ProductName FROM products ORDER BY ProductName ASC`
  - (c) `SELECT ProductName FROM products SORTED BY ProductName ASC`
  - (d) `SELECT ProductName FROM products SORTED BY ProductName DESC`
  - (e) None of the above
23. Which of the following will return a list of every product ID currently listed in the order\_details table where each product ID is listed only once?
- (a) `SELECT DISTINCT ProductID FROM order_details`
  - (b) `SELECT ProductID FROM order_details ONLY ONCE`
  - (c) `SELECT ProductID FROM order_details`
  - (d) `SELECT UNIQUE ProductID FROM order_details`
  - (e) None of the above
24. In the instance of the relation R(A,O,T,V,U) shown below, which of the following functional dependencies hold ?

A O T V U

1 2 3 4 5

1 4 3 4 5

1 2 4 4 1

a)  $A O \rightarrow T; V U \rightarrow A$

b)  $O \rightarrow V; A V \rightarrow U$

c)  $T \rightarrow O; A V \rightarrow U$

d)  $A O \rightarrow T$

e)  $O \rightarrow V; V U \rightarrow A$

25. Which of the following statements contains an error?

(a) `SELECT cid, sum (qty) from orders group by cid having sum(dollars) > 2000;`

(b) `SELECT aid, avg (qty) from orders group by aid;`

(c) `SELECT cid, sum (dollars) from orders;`

(d) `SELECT count (*) from orders;`

26. Which code lists employees by descending order of salary

(a) `SELECT * FROM EMPLOYEES SORT BY SALARY DESCENDING;`

(b) `SELECT * FROM EMPLOYEES IN ORDER OF SALARY;`

(c) `SELECT * FROM EMPLOYEES ORDER BY SALARY DESC;`

(d) `SELECT * FROM EMPLOYEES ORDER BY SALARY;`

27. In order to perform a join, which criteria must be true?

(a) The two tables must have only one column exact same columns.

(b) The tables in the join need to have common rows.

(c) The two tables must both have primary keys

(d) The two tables must have a common column.

28. Consider the follow attributes and functional dependencies:

A B C

$AB \rightarrow C$

$C \rightarrow A$

List all keys (not superkeys):

29. What will result from the following SQL Select statement?



```
Select min(product_description)
```

```
from product_v;
```

- (a) The minimum value of product\_description will be displayed.
- (b) An error message will be generated.
- (c) The first product description alphabetically in product\_v will be shown.
- (d) none of the above

30. The following two SQL statements will produce the same results:

```
Select last_name, first_name
```

```
from customer
```

```
where credit_limit > 99 and credit_limit < 10001;
```

```
Select last_name, first_name
```

```
from customer
```

```
where credit_limit between 100 and 10000;
```

a.TRUE

b.FALSE

31. The following query will execute without errors:

```
select customer.customer_name, salesman.sales_quota
```

```
from customer
```

```
where customer.salesman_id =
```

```
(select salesman_id
```

```
from salesman
```

```
where lname = 'SMITH');
```

a.TRUE

b.FALSE

32. Table TT {J , D , C , V , N , G } and a set of functional dependencies

J,C,N → V,G

D → C,V,G

J → D,C,G

The closure of {D } is:

- a) J D C V N G
- b) J D C V G
- c) J C V G
- d) D C V N
- e) D C V G

33. Table TT {O , M , Z , I , Q } and a set of functional dependencies

$O \rightarrow I, Q$

$I \rightarrow M, Z$

$O, Z \rightarrow M$

Table TT is:

- a) 1 NF
- b) 2 NF
- c) 3 NF

34. For which task in SQL would you use an IN clause

- (a) To query the database for unknown values
- (b) To query the database for a range of values
- (c) To query the database for a character pattern
- (d) To query the database for values in a specified list

35. A Cartesian product is

- (a) A group function
- (b) Produced as a result of a join select statement with no where clause
- (c) The result of fuzzy logic
- (d) A special feature of Oracle Server

36. A type of query that is placed within a WHERE or HAVING clause of another query is called

a:

- (a) master query.
- (b) subquery.
- (c) superquery.
- (d) multi-query.

37. In order to perform a join, which criteria must be true?

- (a) The two tables must have the exact same columns.
  - (b) The tables in the join need to have common rows.
  - (c) The two tables must both have primary keys
  - (d) The two tables must have at least one common column.
38. An attribute(s) which uniquely determines a tuple within a relation is called a:
- (a) Candidate key
  - (b) Foreign key
  - (c) Primary key
  - (d) All of the above
  - (e) A and C
39. Table Q(A,B,C) and a set of functional dependencies
- $AB \rightarrow C;$
- $C \rightarrow A;$
- Table Q is:
- a) 1 NF
  - b) 2 NF
  - c) 3 NF
  - d) BCNF
40. Table Q(A,B,C,D) and a set of functional dependencies
- $AB \rightarrow C;$
- $D \rightarrow B;$
- $C \rightarrow AD$
- Table Q is:
- a) 1 NF
  - e) 2 NF
  - f) 3 NF
  - g) BCNF
41. The following table and functional dependencies exhibits what type of dependency?
- Table(A, B, C)

$A \rightarrow C$

$A \rightarrow B$

$B \rightarrow C$

- (a) Partial Dependence
- (b) Transitive Dependence
- (c) Full Dependence
- (d) A and B
- (e) None of the above

42. In the instance of the relation  $R(A,B,C,D,E)$  shown below, which of the following functional dependencies (FD's) hold? Briefly justify your answer.

A	B	C	D	E
1	2	3	4	5
1	4	3	4	5
1	2	4	4	1

I.  $AB \rightarrow C$       II.  $B \rightarrow D$       III.  $DE \rightarrow B$

- (a) I only
- (b) II only
- (c) I and III only
- (d) II and III only

43. Table  $Q\{A, B, C, D\}$  and a set of functional dependencies

$A \rightarrow B, C$

$D \rightarrow C$

The closure of  $\{A D\}$  is:

- a) A D
- b) A D B C
- c) B C
- d) None of the above

44. Consider the relation  $student(sno, sname, cname, cno)$  where  $(sno, cno)$  or  $(sname, cname)$  are candidate keys. There are functional dependencies within the keys.

The highest normal form whose requirements this relation satisfies is:

- (a) 1NF
- (b) 2NF
- (c) 3NF
- (d) BCNF

45. Let R be a relation with attributes (B,I,N,R,U,L) and let the following functional dependencies hold.

$$B \rightarrow I$$

$$B \rightarrow N$$

$$N R \rightarrow U$$

$$N R \rightarrow L$$

$$I \rightarrow U$$

Given the above functional dependencies, which of the following functional dependencies does not hold:

- a)  $N R \rightarrow U L$
- b)  $B R \rightarrow L$
- c)  $B \rightarrow U$
- d)  $I \rightarrow N R$

46. Suppose relation R(S,G,F,Y,N) has the following functional dependencies:

$$S \rightarrow G$$

$$G \rightarrow F$$

$$G F \rightarrow S$$

$$S \rightarrow Y$$

$$N \rightarrow S$$

$$Y \rightarrow N$$

Which of the following is not a key?

- a) N
- b) G,F
- c) S
- d) Y

47. Table TT {A , N , H , K , J , O , X } and a set of functional dependencies

$$A, N \rightarrow O, X$$

$J \rightarrow A, N, H$  $H, O \rightarrow K, X$  $H, K, X \rightarrow O$  $A, N, X \rightarrow J$ 

The closure of  $\{A, N\}$  is:

e)  $A, H, K, X, J, O$

f)  $A, N, H, K, X, J$

g)  $A, N, H, X, J, O$

h)  $A, N, H, K, X, J, O$

48. Consider the follow attributes and functional dependencies:

 $A, B, C, D, E, F, H$  $A \rightarrow D$  $A, E \rightarrow H$  $D, F \rightarrow B, C$  $E \rightarrow C$  $H \rightarrow E$ 

List all keys (not superkeys):

49. (Đề 48) Consider the decomposition into 3 relations:  $(AD)$   $(EC)$   $(ABEFH)$ . Is this

decomposition in

(a) BCNF

(b) 3NF

(c) 1NF

(d) None of the above

50. Consider a relation  $R(A, B, C)$  with the following functional dependencies:

 $A \rightarrow B; B \rightarrow C$  and  $B \rightarrow A$ 

The number of superkeys of  $R$  is:

(a) 2

(b) 3

(c) 5

(d) 6

**TRƯỜNG ĐẠI HỌC THUẬT CAO THẮNG**  
KHOA ĐT-TH**ĐỀ THI MÔN CƠ SỞ DỮ LIỆU**  
thời gian làm bài 90 phút**CÂU I:(5đ)**

Sau đây là một số lược đồ quan hệ được trích từ bài toán quản lý tuyển sinh:

*ĐIEMTHI(ĐIEMTHISO,ĐIACHIDIEMTHI)*

Một hội đồng coi thi tuyển sinh có nhiều điểm thi, mỗi điểm thi được đặt tại một trường nào đó. Các điểm thi (ĐIEMTHISO) được đánh số là điểm thi số 1, điểm thi số 2, điểm thi số 3,...Mỗi điểm thi xác định địa chỉ (ĐIACHIDIEMTHI).

*THISINH(SOBD,HOTEN,NGAYSINH, PHAI, ĐIACHI, MANGANH, PHONGTHI)*

Mỗi thí sinh có một số báo danh (SOBD) duy nhất, mỗi số báo danh xác định các thông tin: họ và tên (HOTEN), ngày sinh (NGAYSINH), phái (PHAI), địa chỉ thường trú (ĐIACHI), mã ngành đăng ký thi(MANGANH), số hiệu phòng thi(PHONGTHI).

*NGANH(MANGANH,TENNGANH)*

Mỗi ngành có một mã ngành (MANGANH) duy nhất, mỗi mã ngành xác định tên ngành (TENNGANH), chẳng hạn ngành Công Nghệ Thông Tin có mã ngành là 01, ngành Công Nghệ Hoá Thực Phẩm có mã ngành là 10,...

*PHONG(PHONGTHI,ĐIEMTHISO)*

Mỗi điểm thi có nhiều phòng thi (PHONGTHI) được đánh số hiệu khác nhau ở tất cả các điểm thi (trong một phòng thi có thể có các thí sinh của nhiều ngành khác nhau)

1.Xác định khoá cho mỗi lược đồ quan hệ trên (1 điểm)

2.Hãy xác định các ràng buộc toàn vẹn có trong lược đồ cơ sở dữ liệu trên (mỗi loại cho một ví dụ) (1 điểm)

3.Thực hiện các yêu cầu sau bằng SQL (3 điểm)

a.Lập danh sách các thí sinh đăng ký dự thi có số hiệu phòng là "0061", danh sách cần: SOBD,HOTEN,TENNGANH và được sắp tăng dần theo cột SOBD.

b.Danh sách các thí sinh đã đăng ký thi vào ngành có mã ngành là "01", danh sách cần: SOBD, HOTEN, NGAYSINH, PHONGTHI, ĐIACHIDIEMTHI và được sắp tăng dần theo cột SOBD.

c.Hãy thống kê xem mỗi ngành có bao nhiêu thí sinh đã đăng ký thi, danh sách cần: MANGANH,TENNGANH, SOLUONG, trong đó số lượng(SOLUONG) là thuộc tính tự đặt..

**CÂU II: (2đ)**

1. Cho lược đồ quan hệ Q(ABCD), r và s là hai quan hệ được cho như sau:

r				s			
A	B	C	D	A	B	C	D
1	0	0	0	2	1	1	1
1	1	0	0	2	2	1	1
1	1	1	0	1	1	1	0
1	1	1	1	x	y	z	v

Tìm  $r - s$   
 $r * s$

2. Cho hai lược đồ quan hệ Q1(ABC) và Q2(DEF), r và s là hai quan hệ được cho như sau:

r				s		
A	B	C	D	E	F	
1	2	3	1	e	f	
a	b	c	a	e	f	
x	y	z	5	6	7	

A=D

Tìm  $r \bowtie s$

3. Cho hai lược đồ quan hệ Q1(ABC) và Q2(DE), r và s là hai quan hệ được cho như sau

r			s	
A	B	C	D	E
1	2	3	3	1
4	5	6	6	2
7	8	9		

B>D

Tìm  $r \bowtie s$

4. Cho hai lược đồ quan hệ Q<sub>1</sub>(ABCD) và Q<sub>2</sub>(CD), r và s là hai quan hệ được cho như sau:

r				s	
A	B	C	D	C	D
a	b	c	d	c	d



a      b      e      f                      e      f  
 b      c      e      f  
 c      d      e      f  
 a      b      d      e

Tính  $r \div s$

### CÂU III (1đ)

1) Cho lược đồ quan hệ  $Q(ABCDE)$ ,  $r$  là một quan hệ được cho như sau:

A	B	C	D	E
1	2	3	4	5
1	4	3	4	5
1	2	4	4	1

Những phụ thuộc hàm nào sau đây thoả  $r$  ?

$C \rightarrow B$ ;       $AD \rightarrow E$ ;       $B \rightarrow D$ ;       $AB \rightarrow C$ .       $AC \rightarrow D$

2. Cho lược đồ quan hệ  $Q(ABCD)$  và tập phụ thuộc hàm  $F = \{A \rightarrow B ; BC \rightarrow D\}$ .

Những phụ thuộc hàm nào sau đây thuộc  $F^+$  ?

$C \rightarrow D$ ;  $A \rightarrow D$ ;  $AD \rightarrow C$ ;  $AC \rightarrow D$ ;  $BC \rightarrow A$ ;  $B \rightarrow CD$ .

### CÂU IV (2đ)

1. Cho lược đồ quan hệ  $Q$  và tập phụ thuộc hàm  $F$ ,  $K \subseteq F^+$ . Hãy nêu điều kiện để  $K$  là khoá của  $Q$ .

2. Cho lược đồ quan hệ  $Q(ABCD)$  và tập phụ thuộc hàm  $F = \{A \rightarrow D ; D \rightarrow A ; AB \rightarrow C\}$

a. Tính  $AC^+$

b. Tìm tất cả các khoá của  $Q$ .

c.  $Q$  đạt dạng chuẩn nào ? giải thích.

Hết

(Sinh viên không được sử dụng tài liệu

cán bộ coi thi không giải thích gì thêm

## TÀI LIỆU THAM KHẢO

---

---

[1]JEFFREY D. ULLMAN

Principles of database and knowledge base systems

[2].Nguyễn Bá Tường

cơ sở dữ liệu- Lý thuyết và thực hành

[3].NGUYỄN ĐĂNG TỶ- ĐỖ PHÚC

Cơ sở dữ liệu