

BỘ GIÁO DỤC VÀ ĐÀO TẠO
ĐẠI HỌC ĐÀ NẴNG

NGUYỄN THỊ HẢI VY

NGHIÊN CỨU XÂY DỰNG PHẦN MỀM
LẬP LỊCH THI ĐẤU THỂ THAO
TRÊN CƠ SỞ CÁC THUẬT TOÁN ĐỒ THỊ

Chuyên ngành: Khoa học máy tính
Mã số: 60.48.01

TÓM TẮT LUẬN VĂN THẠC SỸ KHOA HỌC

Đà Nẵng - Năm 2011

MỞ ĐẦU

1. Lý do chọn đề tài

Lý thuyết đồ thị là một lĩnh vực nghiên cứu đã có từ lâu và có nhiều ứng dụng trong ngành công nghệ thông tin. Hiện nay sự phát triển của các thuật toán trên đồ thị là một trong các mối quan tâm chính của ngành khoa học máy tính. Việc ứng dụng lý thuyết đồ thị để xây dựng các giải thuật vẫn luôn là vấn đề được nghiên cứu sâu và phát triển cho phù hợp với yêu cầu của công nghệ hiện đại. Từ nhiều bài toán ứng dụng cổ điển của lý thuyết đồ thị, tiếp tục tìm ra các giải pháp cho các bài toán ứng dụng hiện đại là vấn đề cần quan tâm nghiên cứu.

Đề án đổi mới giáo dục đại học Việt Nam đang được thực thi, việc ứng dụng công nghệ thông tin và đổi mới phương pháp giảng dạy, áp dụng phương tiện công nghệ tiên tiến là một trong những nội dung quan trọng trong nền giáo dục hiện nay. Đối với các trường năng khiếu thể thao, việc dạy học và huấn luyện có vài sự khác biệt so với các trường khác nên cách quản lý và giảng dạy cũng khác. Việc ứng dụng công nghệ thông tin từ lâu vẫn luôn được sự quan tâm của lãnh đạo ngành, tuy nhiên chưa có nhiều các phần mềm giảng dạy phục vụ cho các trường năng khiếu thể thao này. Hướng nghiên cứu và kết quả của đề tài nhằm đóng góp một phần vào việc đưa ra giải pháp và thuật toán để xây dựng phần mềm xếp lịch thi đấu thể thao tại các trường năng khiếu TDTT.

Vì vậy, đề tài: “***Nghiên cứu xây dựng phần mềm lập lịch thi đấu thể thao trên cơ sở các thuật toán đồ thị***” sẽ là hướng nghiên cứu tốt để tôi chọn làm đề tài tốt nghiệp thạc sỹ của mình.

2. Mục đích nghiên cứu

Mục đích chính của đề tài là: Nghiên cứu lý thuyết đồ thị và tìm hiểu, nghiên cứu quy trình, cách thức tổ chức các giải thi đấu thể thao. Ứng dụng các thuật toán đồ thị để đề ra giải pháp, thuật toán cho bài toán xếp lịch thi đấu thể thao. Xây dựng, thiết kế phần mềm lập lịch thi đấu thể thao.

3. Đối tượng và phạm vi nghiên cứu

Đối tượng được nghiên cứu cụ thể là: nghiên cứu lý thuyết về đồ thị, các thuật toán trên đồ thị, bài toán tô màu và ứng dụng lập lịch thi, tham khảo một vài hệ thống lập lịch hiện có.

Trong phạm vi giới hạn của đề tài, luận văn nghiên cứu cách lập lịch thi đấu Điền kinh tại các trường năng khiếu TDTT, từ đó ứng dụng triển khai thiết kế và xây dựng phần mềm lập lịch thi đấu Điền kinh biểu diễn trên máy tính dựa trên cơ sở các thuật toán của đồ thị.

4. Phương pháp nghiên cứu

Sử dụng phương pháp nghiên cứu lý thuyết kết hợp với lập trình thực nghiệm.

- Nghiên cứu lý thuyết về Lý thuyết đồ thị, các giải thuật trong đồ thị và ứng dụng, phân tích và tổng hợp tài liệu liên quan.

- Tìm hiểu, nghiên cứu quy trình tổ chức lập lịch thi đấu một giải đấu môn Điền kinh. (Tìm hiểu các ràng buộc của bài toán)

- Phân tích yêu cầu bài toán, xây dựng giải thuật phù hợp.

- Phân tích và thiết kế ứng dụng.

- Xây dựng Demo phần mềm minh họa.

- Đánh giá kết quả theo yêu cầu của đề tài.

5. Ý nghĩa khoa học và thực tiễn của đề tài

Phần nghiên cứu lý thuyết sẽ cung cấp một cách nhìn tổng quát về lý thuyết đồ thị và các thuật toán trong lý thuyết đồ thị.

Kết quả nghiên cứu có thể áp dụng cho các trường năng khiếu thể thao hoặc các Sở văn hóa thể thao du lịch.

6. Cấu trúc của luận văn

Toàn bộ nội dung của Luận văn được chia thành 3 chương như sau:

Chương 1: Trình bày nội dung nghiên cứu tổng quan về lý thuyết đồ thị, một số giải thuật liên quan đến đề tài và giới thiệu vài hệ thống lập lịch hiện có.

Chương 2: Giới thiệu một vài nét về môn thi đấu Điền kinh, mô tả bài toán xếp lịch thi đấu Điền kinh. Phân tích thiết kế hệ thống lập lịch thi đấu môn Điền kinh và giới thiệu một số thuật toán sử dụng để xây dựng phần mềm.

Chương 3: Giới thiệu ứng dụng là phần mềm lập lịch thi đấu môn Điền kinh, cài đặt ứng dụng và đưa ra một số kết quả thực hiện của ứng dụng.

CHƯƠNG 1 – NGHIÊN CỨU TỔNG QUAN

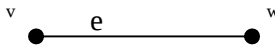
1.1. TỔNG QUAN VỀ LÝ THUYẾT ĐỒ THỊ

1.1.1. Một số khái niệm liên quan đến đồ thị

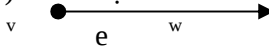
1.1.1.1. Đồ thị, đỉnh, cạnh, cung.

Đồ thị là một cấu trúc rời rạc gồm tập hợp các đỉnh và tập hợp các cạnh nối các đỉnh đó. Người ta phân loại đồ thị dựa trên đặc điểm của các cạnh nối.

- *Đồ thị vô hướng* $G = (V, E)$ gồm một tập V các đỉnh và tập E các cạnh. Mỗi cạnh $e \in E$ được liên kết với một cặp đỉnh v, w (không kể thứ tự)



- *Đồ thị có hướng* $G = (V, E)$ gồm một tập V các đỉnh và tập E các cạnh có hướng gọi là cung. Mỗi cung $e \in E$ được liên kết với một cặp đỉnh (v, w) có thứ tự.



Đồ thị vô hướng có thể coi là đồ thị có hướng trong đó mỗi cạnh $e = (v, w)$ tương ứng với hai cung (v, w) và (w, v) .

+ *Các cạnh song song*: nếu có nhiều cạnh liên kết với cùng một cặp đỉnh.

+ *Khuyên*: là cạnh có hai đỉnh liên kết trùng nhau.

+ *Đỉnh cô lập*: Đỉnh không kề với đỉnh khác.

1.1.1.2. Bậc, nửa bậc vào, nửa bậc ra

Cho đồ thị $G = (V, E)$

- *Bậc của đỉnh* $v \in V$ là tổng số cạnh liên thuộc với nó và ký hiệu là $d(v)$. Nếu đỉnh có khuyên thì mỗi khuyên được tính là 2 khi tính bậc.

- *Nửa bậc*: Cho $G = (V, E)$ là đồ thị có hướng, $v \in V$.

Nửa bậc ra của đỉnh v , ký hiệu là $d_0(v)$, là số cung đi ra từ đỉnh v (v là đỉnh đầu).

Nửa bậc vào của đỉnh $v \in V$, ký hiệu là $d_1(v)$, là số cung đi tới đỉnh v (v là đỉnh cuối).

1.1.1.3. Đường đi, chu trình, tính liên thông

Cho đồ thị $G = (V, E)$. Dây μ từ đỉnh v đến đỉnh w là dãy các đỉnh và cạnh nối tiếp nhau bắt đầu từ đỉnh v và kết thúc tại đỉnh w . Số cạnh trên dây μ gọi là *độ dài* của dây μ .

Dây μ từ đỉnh v đến đỉnh w độ dài n được biểu diễn như sau:

$$\mu = (v, e_1, v_1, e_2, v_2, \dots, v_{n-1}, e_n, w)$$

trong đó v_i ($i = 1, \dots, n-1$) là các đỉnh trên dây và e_i ($i = 1, \dots, n$) là các cạnh trên dây liên thuộc đỉnh kề trước và sau nó. Các đỉnh và cạnh trên dây có thể lặp lại.

- *Đường đi* từ đỉnh v đến đỉnh w là dây từ đỉnh v đến đỉnh w , trong đó các cạnh không lặp lại.

- *Chu trình* là đường đi có đỉnh đầu và đỉnh cuối trùng nhau.

- *Đường đi có hướng* trong đồ thị có hướng là dây có hướng, trong đó các cung không lặp lại.

Đồ thị vô hướng gọi là *liên thông*, nếu mọi cặp đỉnh của nó đều có đường đi nối chúng với nhau.

Đồ thị có hướng gọi là *liên thông mạnh*, nếu mọi cặp đỉnh của nó đều có đường đi có hướng nối chúng với nhau.

1.1.2. Biểu diễn đồ thị trên máy tính

1.1.2.1. Biểu diễn đồ thị bằng ma trận kề

1.1.2.2. Biểu diễn đồ thị bằng ma trận liên thuộc

1.1.2.3. Biểu diễn bằng danh sách kề

1.1.3. Đồ thị đẳng cấu

1.1.4. Đồ thị phẳng

1.2. MỘT SỐ GIẢI THUẬT LIÊN QUAN ĐẾN ĐỀ TÀI

1.2.1. Tô màu bản đồ

1.2.2. Thuật toán tuần tự ưu tiên đỉnh bậc lớn nhất

Cho đồ thị $G=(V,E)$. Thuật toán sau sẽ tô màu các đỉnh đồ thị với số màu k gần với sắc số $\chi(G)$.

(i) Lập danh sách các đỉnh đồ thị $E' := [v_1, v_2, \dots, v_n]$

theo thứ tự bậc giảm dần $d(v_1) \geq d(v_2) \geq \dots \geq d(v_n)$

Đặt $i := 1$

(ii) Tô màu i cho đỉnh đầu tiên trong danh sách. Duyệt lần lượt các đỉnh tiếp theo và tô màu i cho đỉnh không kề đỉnh đã được tô màu i

(iii) Nếu tất cả các đỉnh đã được tô màu thì kết thúc: Đồ thị đã được tô màu bằng i màu. Ngược lại sang bước (iv)

(iv) Loại khỏi E' các đỉnh đã tô màu, đặt $i := i + 1$, và quay lại bước (ii).

1.2.3. Tô màu đồ thị phẳng

1.3. MỘT SỐ BÀI TOÁN ỨNG DỤNG

1.3.1. Bài toán lập lịch thi

Giả sử mỗi sinh viên phải thi một số môn trong n môn thi. Hãy lập lịch thi trong trường đại học sao cho không có sinh viên nào thi hai môn cùng một thời gian và số đợt thi là ít nhất.

Giải pháp:

Bài toán được biểu diễn bằng đồ thị:

- Mỗi môn thi là một đỉnh.

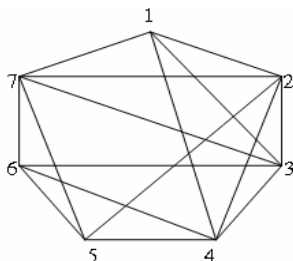
- Nếu 2 môn thi nào được dự thi bởi cùng một sinh viên thì sẽ được nối bằng một cạnh.

- Cách lập lịch thi sẽ tương ứng với bài toán tô màu của đồ thị này.

Ví dụ. Có 7 môn thi cần xếp lịch. Các môn học được đánh số từ 1 đến 7. Các cặp môn thi sau có chung sinh viên:

(1,2), (1,3), (1,4), (1,7), (2,3),
 (2,4), (2,5), (2,7), (3,4),(3,6),
 (3,7),(4,5),(4,6),(5,7),(6,7),(5,6)

Đồ thị tương ứng như sau:



Hình 1.6 – Đồ thị minh họa bài toán lập lịch thi

Áp dụng thuật toán ta tô màu các đỉnh đồ thị có được kết quả sau:

Đỉnh:	2	3	4	7	1	5	6
Bậc:	5	5	5	5	4	4	4
Màu:	1	2	3	3	4	2	1

Như vậy ta có lịch thi gồm 4 đợt thi như sau:

Đợt 1: Thi các môn 2, 6

Đợt 2: Thi các môn 3, 5

Đợt 3: Thi các môn 4, 7

Đợt 4: Thi môn 1

1.3.2. Bài toán phân chia tần số

1.3.3. Bài toán điều khiển đèn hiệu nút giao thông

1.4. ĐÁNH GIÁ MỘT VÀI HỆ THỐNG LẬP LỊCH HIỆN CÓ

1.4.1. Lịch trình thi đấu bóng đá “2010 World Cup Final Tournament Schedule”

1.4.2. Bài toán tô màu và ứng dụng xây dựng phần mềm xếp lịch thi cho học chế tín chỉ

1.4.3. Bài toán tạo lịch thi đấu Tennis theo thuật toán chia để trị.

1.4.4. Bài toán xếp lịch thi đấu môn Bóng đá theo thuật toán chia để trị.

CHƯƠNG 2 - PHÂN TÍCH THIẾT KẾ HỆ THỐNG LẬP LỊCH THI ĐẤU MÔN ĐIỀN KINH

2.1. TỔNG QUAN VỀ MÔN ĐIỀN KINH

2.1.1 Điền kinh là gì?

2.1.2. Điều lệ thi đấu một giải đấu điền kinh

2.1.3. Quy trình lập lịch thi đấu truyền thống

2.2. MÔ TẢ BÀI TOÁN XẾP LỊCH THI ĐẤU ĐIỀN KINH

2.2.1. Đặc tả bài toán

Điền kinh là một môn thể thao bao gồm nhiều nội dung thi đấu (chạy, nhảy, đá cầu, đẩy tạ...). Một giải đấu điền kinh được tổ chức thi theo các nội dung thi. Nếu vận động viên đáp ứng tốt các yêu cầu của điều lệ thi đấu có thể đăng ký tham gia giải đấu điền kinh. Ban tổ chức giải đấu sẽ thông báo các nội dung thi để các vận động viên có thể đăng ký tham gia. Sau khi có bảng danh sách chính

thức về các vận động viên đăng ký, ban tổ chức sẽ tiến hành lập lịch thi đấu cho toàn giải.

Như vậy, một vận động viên có thể đăng ký tối đa n nội dung thi đấu (n là số lượng nội dung thi mà một vận động viên được tham gia theo qui định của điều lệ giải). Vì vậy, lịch thi cần phải được bố trí để nếu có một vận động viên đăng ký nhiều nội dung thi đấu thì các nội dung thi này không được thi cùng ngày.

Lịch thi được chia thành nhiều ngày thi. Trong một ngày, một địa điểm (sân vận động) có thể tổ chức nhiều nội dung thi.

Mỗi vận động viên sẽ thi một nội dung trong một ngày thi. Mỗi nội dung có thể thi trong một ngày hoặc nhiều ngày (tùy giải đấu).

2.2.2. Xây dựng đồ thị cho bài toán

Cho đồ thị N đỉnh ($N \leq 50$), mỗi đỉnh là một nội dung thi đấu của môn Điền kinh.

Nếu 2 nội dung thi đấu nào được dự thi bởi cùng một vận động viên thì sẽ được nối bằng một cạnh.

Cách lập lịch thi đấu sẽ tương ứng với bài toán tô màu của đồ thị này.

2.3. PHÂN TÍCH YÊU CẦU BÀI TOÁN VÀ THIẾT KẾ CƠ SỞ DỮ LIỆU

2.3.1. Phân tích yêu cầu bài toán

2.3.1.1. Dữ liệu đầu vào

- Dữ liệu các thông tin về giải đấu Điền kinh.
- Danh sách các vận động viên đăng ký thi.
- Những nội dung tổ chức thi đấu.
- Số môn thi tổ chức trong ngày, tùy giải đấu lớn nhỏ mà qui mô tổ chức và cách thức bố trí lịch thi có nhiều môn thi khác nhau.

- Thời gian thi đấu: từ ngày ... đến ngày.....

2.3.1.2. Các ràng buộc của bài toán

- Số ngày thi là ít nhất.
- Phân bổ các vận động viên thi đấu vào các ngày thi sao cho chúng không xung đột nhau, vận động viên dự thi không trùng lịch (không vi phạm ràng buộc là nếu có vận động viên thi nhiều hơn 1 nội dung thì các nội dung thi này không được thi cùng ngày với nhau).
- Vận động viên thi đấu theo giới tính riêng, không có nội dung hỗn hợp nam nữ.

2.3.1.3. Dữ liệu đầu ra

Lịch thi gồm:

- Ngày thi
- Nội dung thi
- Địa điểm thi
- Thông tin chi tiết VĐV dự thi ở từng nội dung.

2.3.2. Phân tích hệ thống và thiết kế cơ sở dữ liệu:

2.3.2.1. Xác định các thực thể

1. Thực thể **tblTest:**

- Chứa các thông tin chi tiết về các nội dung thi đấu trong giải đấu Điền kinh.
- Các thuộc tính: testCode, NameTest, Description
- Mô tả: Mỗi nội dung thi đấu có một mã nội dung (testCode) duy nhất, mỗi testCode xác định một tên nội dung thi đấu (NameTest), một mô tả cụ thể về nội dung thi đấu đó (Description).

2. Thực thể **tblAtheletes:**

- Chứa thông tin về vận động viên đăng ký tham gia thi đấu.

- Các thuộc tính: athletesCode, athletesName, Sex, Birthday, Team, Address, Tel.

- Mô tả: Mỗi vận động viên đăng ký tham gia thi đấu có một mã vận động viên duy nhất (athletesCode), mỗi athletesCode xác định tên một vận động viên (athletesName), giới tính vận động viên (Sex), ngày sinh (Birthday), thuộc đội thi nào (Team), địa chỉ vận động viên (Address), số điện thoại liên lạc (Tel).

3. Thực thể **tblRegister**:

- Ghi nhận các thông tin đăng ký dự thi của vận động viên.
- Các thuộc tính: athletesCode, testCode, ExamCode.
- Mô tả: Thông qua mã vận động viên (athletesCode), mã nội dung thi đấu (testCode), mã kỳ thi (ExamCode), mỗi vận động viên có thể tham gia đăng ký một hoặc nhiều nội dung thi đấu trong một kỳ thi bất kỳ.

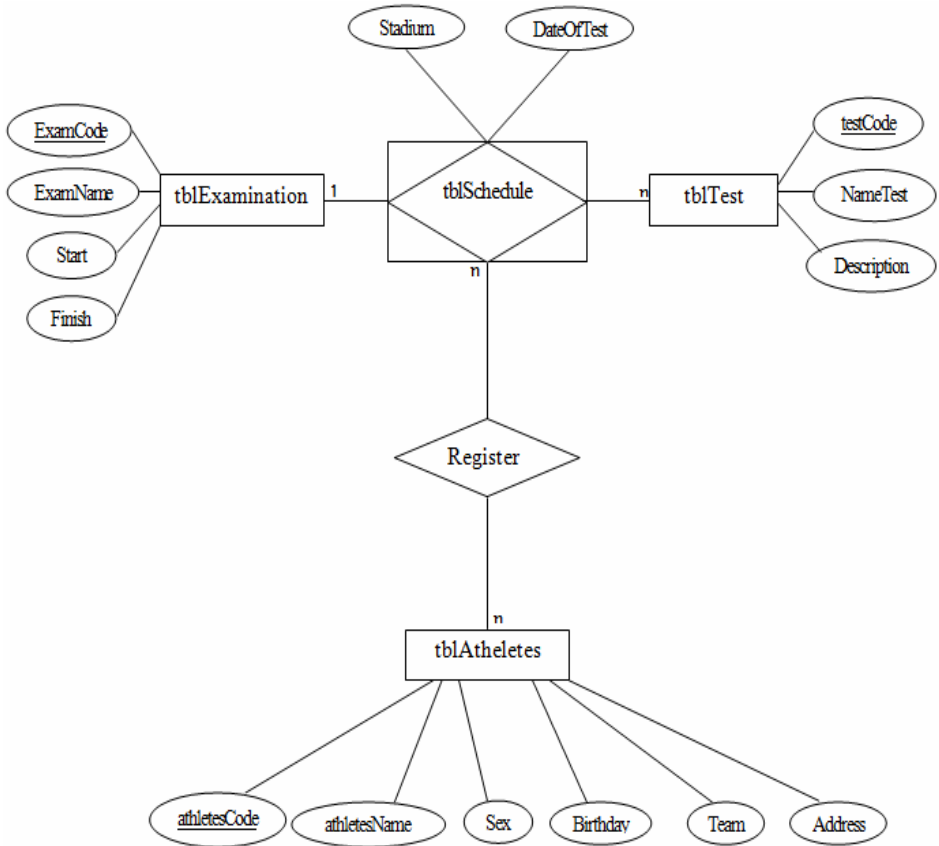
4. Thực thể **tblSchedule**:

- Mô tả chi tiết các thông tin về lịch trình thi đấu.
- Các thuộc tính: ExamCode, testCode, DateofTest, Stadium.
- Mô tả: Thực thể này sẽ trả về kết quả của lịch thi là mỗi kỳ thi (thông qua mã kỳ thi ExamCode) sẽ có các nội dung thi (thông qua mã nội dung thi testCode), các nội dung thi đấu sẽ được phân chia vào từng ngày thi tương ứng (DateofTest), thi tại sân vận động nào (Stadium)

5. Thực thể **tblExamination**:

- Chứa các thông tin về kỳ thi đấu Điền kinh.
- Các thuộc tính: ExamCode, ExamName, Start, Finish.
- Mô tả: Mỗi kỳ thi đấu tương ứng với một mã kỳ thi (ExamCode) duy nhất, mỗi ExamName xác định một tên kỳ thi, thời gian bắt đầu (Start) và thời gian kết thúc kỳ thi (Finish) của kỳ thi đó.

2.3.2.2. Mô hình thực thể quan hệ



Hình 2.1- Mô hình thực thể quan hệ

Trong mô hình này biểu diễn một kỳ thi đấu Điền kinh bắt kỳ (tblExamination) sẽ tổ chức nhiều nội dung thi đấu (tblTest). Một vận động viên (tblAtheletes) sẽ tham gia đăng ký các nội dung thi trong kỳ thi này. Sau khi có các thông tin cần thiết, ban tổ chức kỳ thi sẽ tạo ra một lịch thi đấu cho các vận động viên tham gia. Kết quả của lịch thi sẽ thể hiện rõ vận động viên thi nội dung nào thuộc ngày thi nào, tại sân vận động nào.

2.3.2.3. Các ràng buộc toàn vẹn dữ liệu

Ta có cơ sở dữ liệu TDDT như sau:

tblExamination (ExamCode, ExamName, Start, Finish)

tblTest (testCode, NameTest, Description)

tblAtheletes (athletesCode, athletesName, Sex, Birthday, Team, Address, Tel)

tblRegister (ExamCode, testCode, athletesCode)

tblSchedule (ExamCode, testCode, DateOfTest, Stadium)

a. Toàn vẹn thực thể:

Qui tắc toàn vẹn thực thể yêu cầu mỗi bảng quan hệ (thực thể) phải có khóa chính, các thuộc tính khóa phải có giá trị duy nhất và khác *null*. Qui tắc này không cho phép hai bản ghi trùng khóa.

Trong bài viết này, ta có các qui tắc toàn vẹn thực thể ràng buộc sau:

- Trong quan hệ **tblExamination**, thuộc tính khóa *ExamCode* là khóa chính nên không thể nhận giá trị *null* và có giá trị không trùng nhau, không có khoảng trắng.

- Trong quan hệ **tblTest**, thuộc tính khóa *testCode* là khóa chính nên không thể nhận giá trị *null* và có giá trị không trùng nhau, không có khoảng trắng.

- Trong quan hệ **tblAtheletes**, thuộc tính khóa *athletesCode* là khóa chính nên không thể nhận giá trị *null* và có giá trị không trùng nhau, không có khoảng trắng.

- Trong quan hệ **tblRegister**, các thuộc tính khóa (*ExamCode*, *testCode*, *athletesCode*) không thể nhận giá trị *null* và có giá trị không trùng nhau.

- Trong quan hệ **tblSchedule**, cặp thuộc tính khóa (*ExamCode*, *testCode*) không thể nhận giá trị *null* và có giá trị không trùng nhau.

b. Toàn vẹn tham chiếu

Toàn vẹn tham chiếu là ràng buộc đảm bảo tính hợp lệ của sự tham chiếu của một đối tượng trong cơ sở dữ liệu (gọi là đối tượng tham chiếu) đến đối tượng khác (gọi là đối tượng được tham chiếu) trong cơ sở dữ liệu đó. Các thuộc tính tương ứng gọi là *thuộc tính cặp ghép* của ràng buộc tham chiếu.

Quy tắc toàn vẹn tham chiếu được xét đến trong khi cập nhật quan hệ tham chiếu hoặc quan hệ được tham chiếu.

Ta có các toàn vẹn tham chiếu sau:

- Thuộc tính *athletesCode* của thực thể **tblRegister** là khóa ngoại tham chiếu đến khóa chính *athletesCode* của quan hệ **tblAtheletes**:

tblRegister.athletesCode → *tblAtheletes.athletesCode*

Quan hệ **tblRegister** là quan hệ tham chiếu và quan hệ **tblAtheletes** là quan hệ được tham chiếu, với cặp ghép (*tblRegister.athletesCode*, *tblAtheletes.athletesCode*)

- Thuộc tính *testCode* của thực thể **tblRegister** là khóa ngoại tham chiếu đến khóa chính *testCode* của quan hệ **tblTest**:

tblRegister.testCode → *tblTest.testCode*

Quan hệ **tblRegister** là quan hệ tham chiếu và quan hệ **tblTest** là quan hệ được tham chiếu, với cặp ghép (*tblRegister.testCode* , *tblTest.testCode*)

- Thuộc tính *ExamCode* của thực thể **tblRegister** là khóa ngoại tham chiếu đến khóa chính *ExamCode* của quan hệ **tblExamination**:

tblRegister.ExamCode → *tblExamination.ExamCode*

Quan hệ **tblRegister** là quan hệ tham chiếu và quan hệ **tblExamination** là quan hệ được tham chiếu, với cặp ghép (*tblRegister.ExamCode*, *tblExamination.ExamCode*)

- Thuộc tính *testCode* của thực thể **tblSchedule** là khóa ngoại tham chiếu đến khóa chính *testCode* của quan hệ **tblTest**:

tblSchedule.testCode → *tblTest.testCode*

Quan hệ **tblSchedule** là quan hệ tham chiếu và quan hệ **tblTest** là quan hệ được tham chiếu, với cặp ghép (*tblSchedule.testCode*, *tblTest.testCode*)

- Thuộc tính *ExamCode* của thực thể **tblSchedule** là khóa ngoại tham chiếu đến khóa chính *ExamCode* của quan hệ **tblExamination**:

tblSchedule.ExamCode → *tblExamination.ExamCode*

Quan hệ **tblSchedule** là quan hệ tham chiếu và quan hệ **tblExamination** là quan hệ được tham chiếu, với cặp ghép (*tblSchedule.ExamCode*, *tblExamination.ExamCode*)

c. Miền giá trị:

Đây là loại ràng buộc lên các giá trị hợp lệ của thuộc tính. Miền giá trị là tập hợp tất cả các loại dữ liệu và phạm vi giá trị được thuộc tính thừa nhận. Trong cơ sở dữ liệu này, miền giá trị xác định các tham số đặc trưng của các thuộc tính như sau:

Xét quan hệ **tblTest** (*testCode*, *NameTest*, *Description*)

Tên	Ý nghĩa	Kiểu dữ liệu	Độ dài	Null support
<i>testCode</i>	Mã nội dung thi	Ký tự (nchar)	10	Non-null
<i>NameTest</i>	Tên nội dung thi đầu	Ký tự (nvarchar)	50	Non-null
<i>Description</i>	Mô tả chi tiết	nvarchar	50	Null

Xét quan hệ: **tblAtheletes** (*athletesCode*, *athletesName*, *Sex*, *Birthday*, *Team*, *Address*, *Tel*).

Tên	Ý nghĩa	Kiểu dữ liệu	Độ dài	Khuôn dạng	Phạm vi	Null support
<i>athletesCode</i>	Mã vận động viên	Ký tự (nchar)	10			Non-null
<i>athletesName</i>	Tên vận động viên	Ký tự (nvarchar)	50			Non-null
<i>Sex</i>	Giới tính	Ký tự	1	'-' nữ '+' nam '0' đồng tính		Non-null
<i>Birthday</i>	Ngày sinh	smalldatetime			< Ngày hiện hành	Non-null
<i>Team</i>	Đội thi	nvarchar	50			Non-null
<i>Address</i>	Địa chỉ	nvarchar	50			Non-null
<i>Tel</i>	Điện thoại	Ký tự (nchar)	12			Null

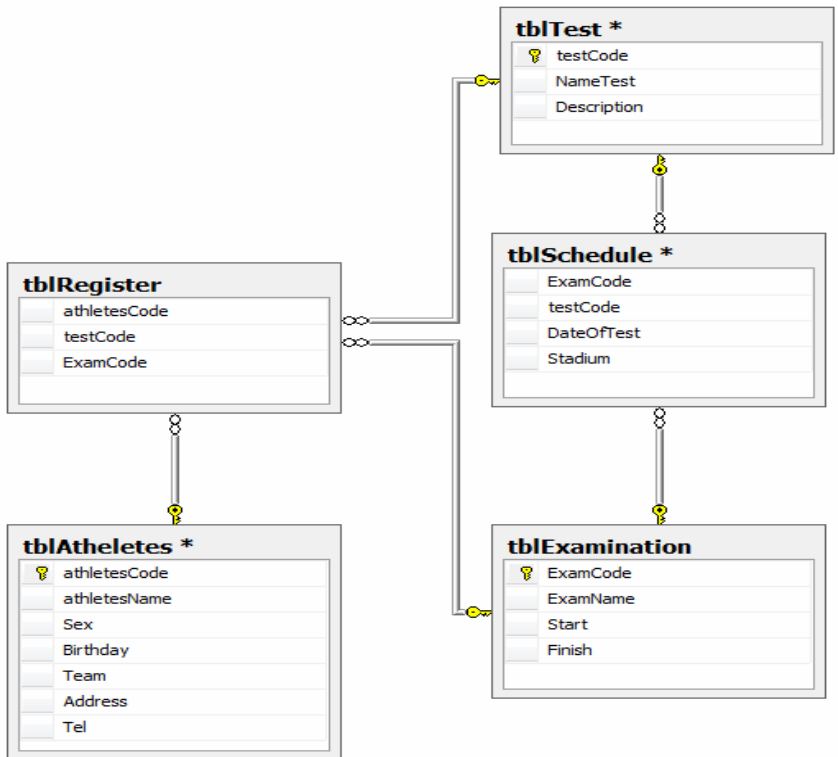
Xét quan hệ **tblExamination** (*ExamCode*, *ExamName*, *Start*, *Finish*)

Tên	Ý nghĩa	Kiểu dữ liệu	Độ dài	Phạm vi	Null support
<i>ExamCode</i>	Mã kỳ thi	Ký tự (nchar)	10		Non-null
<i>ExamName</i>	Tên kỳ thi	Ký tự (nvarchar)	50		Non-null
<i>Start</i>	Ngày bắt đầu	datetime			Non-null
<i>Finish</i>	Ngày kết thúc	datetime		$\geq$ <i>Start</i>	Non-null

Xét quan hệ **tblSchedule** (ExamCode, testCode, DateOfTest, Stadium)

Tên	Ý nghĩa	Kiểu dữ liệu	Độ dài	Phạm vi	Null support
<i>DateOfTest</i>	Ngày thi	smalldatetime		<i>Start</i> <= <i>DateOfTest</i> <= <i>Finish</i>	Non-null
<i>Stadium</i>	Sân vận động	nvarchar	50		Null

2.3.2.4. Mô hình cơ sở dữ liệu được cài đặt như sau



Hình 2.2- Mô hình cơ sở dữ liệu

2.4. CÁC THUẬT TOÁN SỬ DỤNG XÂY DỰNG PHẦN MỀM

2.4.1. Giải thuật tựa ngôn ngữ

- Nhập dữ liệu đầu vào:

+ dsVDV: danh sách vận động viên đăng ký thi các nội dung.

+ dsNDthi: là danh sách các nội dung thi đấu

+ dsExam: danh sách các giải đấu.

- Đầu ra:

+ SoNgay: số lượng ngày cần thiết tối ưu để tổ chức thi đấu.

+ MonThi: nội dung thi đấu được tổ chức tại giải đấu.

+ dsSchedule: là danh sách lịch thi đấu tương ứng với thông tin ngày thi, nội dung thi và số lượng vận động viên dự thi.

Tư tưởng giải thuật:

- Bước 1: Tính bậc cho các đỉnh trên đồ thị và sắp xếp các đỉnh này theo thứ tự bậc giảm dần vào một mảng.

Đặt $i = 1$;

- Bước 2: Tô màu i cho đỉnh đầu tiên có bậc cao nhất trong mảng đã sắp xếp ở bước 1. Duyệt lần lượt các đỉnh tiếp theo và tô màu i cho đỉnh không kề với đỉnh đã tô màu i .

- Bước 3: Nếu tất cả các đỉnh đã được tô màu thì kết thúc, đồ thị được tô bằng i màu. Ngược lại, sang bước 4.

- Bước 4: Loại ra khỏi mảng các đỉnh đã tô màu, sắp xếp lại các đỉnh theo thứ tự bậc giảm dần. Đặt $i = i + 1$ và quay lại bước 2.

2.4.2. Các bước giải thuật

- *Bước 1:* Xây dựng một đồ thị với n đỉnh, tập đỉnh ở đây chính là các nội dung thi đấu và mối quan hệ của các đỉnh này dựa vào thông tin đăng ký dự thi đầu vào của các vận động viên.

$n = dsLLich$: số lượng đỉnh của đồ thị

`arrSoThuTu[]`: mảng chứa các đỉnh theo thứ tự bậc giảm dần.

`mau = cacMon[arrSoThuTu[i]].MauDaTo` : đỉnh i có bậc cao nhất của đồ thị được tô màu là *mau*, đỉnh i tương ứng với nội dung thi đầu i . Nếu *mau* nhận giá trị 0 tức là nội dung thi đầu i chưa được tô màu.

- *Bước 2*: Áp dụng bài toán tô màu trong lập lịch thi đầu

+ Khởi tạo cho màu hiện hành bằng 0

```
this.cacMon[i].MauDaTo = 0;
```

1. Tính bậc cho các đỉnh trên đồ thị:

```
decimal[] arrBac = new decimal[cacMon.Length];
```

```
arrBac[i] = Convert.ToDecimal(cacMon[i].lstCacMon.Count);
```

2. Sắp xếp các đỉnh vào mảng theo thứ tự bậc giảm dần:

```
int[] arrSoThuTu = new int[cacMon.Length];
```

```
quickSort(ref arrSoThuTu, ref arrBac, 0, cacMon.Length - 1);
```

3. Tô màu cho đỉnh đầu tiên có bậc cao nhất trong mảng đã sắp xếp:

```
int mau = 1;
```

```
for (int i = 0; i < arrSoThuTu.Length; i++)
```

```
{
```

```
    if (i == 0)
```

```
{
```

```
        cacMon[arrSoThuTu[i]].MauDaTo = mau;
```

```
    }
```

```
.....
```

4. Duyệt lần lượt các đỉnh tiếp theo và tìm đỉnh j để tô màu giống i thỏa mãn điều kiện đỉnh j có số bậc lớn nhất mà không kề với i .

```
    mau = 0;
```

```
    for (int j = 0; j < i; j++)
```

```

{
    if
(cacMon[arrSoThuTu[i]].lstCacLangGieng.Contains(cacMon[
arrSoThuTu[j]].Code))
        mau = cacMon[arrSoThuTu[j]].MauDaTo;
    else
        break;
}
cacMon[arrSoThuTu[i]].MauDaTo = ++mau;

```

Trong bước 2 này, ta phải xây dựng một số hàm như:

- + Tính bậc cho các đỉnh trên đồ thị.
- + Hàm sắp xếp các đỉnh theo thứ tự bậc giảm dần.

```

private void quickSort(ref int[] arrSoThuTu, ref decimal[] arrBac, int
batDau, int ketThuc)

```

- *Bước 3*: Lập lịch thi: thuật toán dựa trên các nội dung thi tương ứng với từng màu thu được.

Xử lý yêu cầu bài toán lập lịch:

- * Chọn các nội dung thi đấu trong cơ sở dữ liệu cần lập lịch
- * Nhập thông tin tối thiểu để lập lịch
- * Nhập thông tin về thời gian bắt đầu cho một đợt thi

- *Bước 4*: Hiển thị kết quả xếp lịch thi

- * Chuẩn hóa tham số đầu ra: SoNgay, Monthi, Ngaythi
- * Cập nhật thời gian thi tương ứng với các nội dung thi đấu.
- * Chuẩn hóa dữ liệu thành lịch thi tương ứng

CHƯƠNG 3 - XÂY DỰNG PHẦN MỀM LẬP LỊCH THI ĐẤU MÔN ĐIỀN KINH

3.1. GIỚI THIỆU CÔNG CỤ XÂY DỰNG PHẦN MỀM

3.1.1. Hệ quản trị cơ sở dữ liệu SQL Server 2005

3.1.2. Ngôn ngữ lập trình C#

3.2. CÀI ĐẶT CÁC MODULE CHỨC NĂNG

Bảng 3.1- Bảng danh sách các module chức năng

TT	Module	Chức năng
1	VDVDangKyDuThi	Cập nhật dữ liệu về vận động viên đăng ký các nội dung thi đấu Điền kinh. - Cập nhật đăng ký mới. - Xóa đăng ký. - Lưu trữ và xử lý các thông tin cá nhân của sinh viên.
2	QlyNoiDungThiDau	Quản lý các thông tin về nội dung thi đấu - Lưu trữ các thông tin về nội dung thi đấu - Thêm mới các nội dung thi - Cập nhật thông tin về nội dung thi - Xóa đăng ký nội dung
3	Laplichthi	Xử lý yêu cầu của bài toán lập lịch: - Cho phép chọn các nội dung thi đấu trong CSDL nội dung thi cần lập lịch - Cho phép tạo một kỳ thi mới. - VĐV đăng ký môn thi. - Nhập số đợt thi muốn tổ chức trong ngày - Nhập thông tin về thời gian bắt đầu và kết thúc cho mỗi đợt thi.
4	KetQuaLaplichthi	Hiển thị kết quả đã lập lịch: - Thống kê danh sách các nội dung thi sẽ tổ chức thi đấu, tương

		ứng mỗi nội dung thi sẽ là danh sách các VĐV tham gia thi đấu nội dung đó. - Tổng kết chung về kết quả lập lịch: Môn thi nào tương ứng thi đợt nào, ngày thi nào, thi trong vòng bao nhiêu ngày.
5	TimKiemThongtin VDV	Xử lý và hiển thị kết quả tra cứu và tìm kiếm thông tin VĐV dự thi.
6	TimKiemThongtin NoiDungthi	Xử lý và hiển thị kết quả tra cứu và tìm kiếm thông tin các nội dung sẽ tổ chức thi đấu trong kỳ thi.

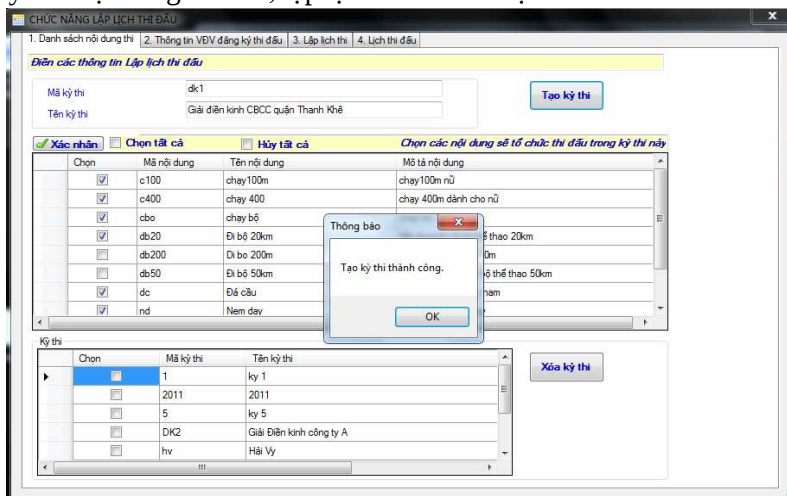
3.3. CÀI ĐẶT MỘT SỐ CODE CHO THUẬT TOÁN

3.4. MỘT SỐ GIAO DIỆN DEMO PHẦN MỀM

3.4.1. Giao diện chính

3.4.2. Chức năng quản lý thông tin

3.4.3. Chức năng lập lịch: là chức năng chính của demo phần mềm ứng dụng bài toán tô màu, bao gồm việc lên danh sách các nội dung thi đấu, tạo một kỳ thi đấu bất kỳ, vận động viên đăng ký các nội dung thi đấu, lập lịch thi và xem lịch thi.



Hình 3.4 – Giao diện tạo kỳ thi đấu

Một vận động viên có thể đăng ký thi đấu nhiều môn thi:

CHỨC NĂNG LẬP LỊCH THI ĐẤU

1. Danh sách nội dung thi | 2. Thông tin VĐV đăng ký thi đấu | 3. Lập lịch thi | 4. Lịch thi đấu

Mã VĐV	Tên VĐV	Giới tính	Ngày sinh	Đội	Địa chỉ	Số điện thoại
hvv	Nguyễn Thị Hải Vy	Nữ	10/11/1983	Trường TDTT Đà...	Đà Nẵng	905054289
hvb	Hồ Văn Ba	Nữ	01/12/1980	TDTTQN	Quảng Ninh	983777111
ntb	Nguyễn Thị Bé	Nữ	18/01/1983	Đoàn TDTT Hải ...	Hải Phòng	909444555
nvp	Nguyễn Văn Phúc	Nam	14/10/1981	Đoàn TDTT QNinh	Quảng Ninh	909999333
pql	Phan Quang Lâm	Nam	02/01/2011	TDTT Đồng Nai	Đồng Nai	998173456
tod	Trương Công Đình	Nam	05/02/1983	Sở TDTT Hà Tĩnh	Hà Tĩnh	1999890444
tdđ	Trần Đình Dũng	Nữ	26/02/1996	TDTTLC	Lai Châu	905334587
tdc	Trần Đình Dũng	Nam	26/02/1996	TDTTLC	Lai Châu	905334587
tdh	Trương Đình Hà	Nam	01/12/1980	TDTTQN	Quảng Ninh	983777111

Kỳ thi: Giải điền kinh CBCC quận Thanh Khê

Mã VĐV: **hvv**

Tên VĐV: **Nguyễn Thị Hải Vy**

Thông báo: Đăng ký môn thi thành công.

Đăng ký môn thi

Chọn	Mã nội dung	Tên nội dung
☒	c100	chạy 100m
☐	c400	chạy 400
☒	cbo	chạy bộ
☐	db20	Đi bộ 20km
☒	dc	Đá cầu
☐	nd	Ném dĩa

Hình 3.5 – Giao diện đăng ký môn thi đấu

CHỨC NĂNG LẬP LỊCH THI ĐẤU

1. Danh sách nội dung thi | 2. Thông tin VĐV đăng ký thi đấu | 3. Lập lịch thi | 4. Lịch thi đấu

Kỳ thi: Giải điền kinh CBCC quận Thanh Khê

Lịch thi từ ngày: 25/07/2011 Đến ngày: 27/07/2011

Địa điểm thi: Chi Lăng

Lập lịch

Thông báo: Lập lịch thi thành công.

Hình 3.6 – Giao diện lập lịch thi đấu thành công

Giao diện kết quả lập lịch:

CHỨC NĂNG LẬP LỊCH THI ĐẤU

1. Danh sách nội dung thi | 2. Thông tin VĐV đăng ký thi đấu | 3. **Lập lịch thi** | 4. Lịch thi đấu

Kỳ thi: Giải điền kinh CBBCC quận Thanh Khê

Tên nội dung	Ngày thi	Sân vận động
Điền kinh	25/07/2011	Chi Lăng
chạy 400	26/07/2011	Chi Lăng
chạy 100m	27/07/2011	Chi Lăng
chạy 800	28/07/2011	Chi Lăng
Ném dĩa	28/07/2011	Chi Lăng
Đá cầu	29/07/2011	Chi Lăng
Nhảy xa	29/07/2011	Chi Lăng

Vận động viên:

Mã VĐV	Tên VĐV	Giới tính	Ngày sinh	Đội	Địa chỉ	Số điện thoại
hnh	Hồ Văn Ba	Nữ	01/12/1980	TDTGN	Quảng Ninh	983777111
ntb	Nguyễn Thị Bé	Nữ	18/01/1983	Đoàn TDTT Hải ...	Hải Phòng	909444555
nvp	Nguyễn Văn Phúc	Nam	14/10/1981	Đoàn TDTT QNinh	Quảng Ninh	909999333
pql	Phan Quang Lâm	Nam	02/01/2011	TDTT Đồng Nai	Đồng Nai	998173456
tde	Trần Đình Dũng	Nam	26/02/1996	TDTTLC	Lai Châu	905334587

Hình 3.7 – Giao diện kết quả lập lịch

3.4.4. Chức năng tìm kiếm thông tin

3.5. ĐÁNH GIÁ PHẦN MỀM

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Luận văn đã nghiên cứu và trình bày những kiến thức căn bản về lý thuyết đồ thị và qui trình lập lịch thi đấu Điền kinh dựa trên thuật toán đồ thị. Qua đó luận văn đạt được một số kết quả như sau:

Về lý thuyết, luận văn đã đi sâu nghiên cứu được các kiến thức chung về lý thuyết đồ thị và ứng dụng, nêu được chi tiết các thuật toán trên đồ thị, tìm hiểu một vài hệ thống lập lịch hiện có, dựa vào đó triển khai ứng dụng của thuật toán tô màu đồ thị cho bài toán lập lịch thi đấu Điền kinh.

Về ứng dụng minh họa, với mục tiêu làm rõ thêm lý thuyết, luận văn ứng dụng xây dựng phần mềm lập lịch thi đấu Điền kinh. Từ kết quả tìm hiểu về cách thức lập lịch thi đấu, bài toán xếp lịch thi đấu được quy về bài toán tô màu đồ thị với thuật toán tuần tự ưu tiên đỉnh bậc lớn nhất.