# LỜI CẢM ƠN

Chúng em xin chân thành cảm ơn thầy Trương Hải Bằng, người thầy đã tận tình giúp đỡ chúng em trong suốt quá trình làm đề tài.

Trong quá trình thực hiện đề tài, chúng em không tránh khỏi những sai sót trong quá trình tìm hiểu và thực hiện, chúng em mong thầy xem qua báo cáo và góp ý để giúp em có kinh nghiệm trong việc tìm hiểu và phát triển bài toán tô màu và ứng dụng.

Một lần nữa, chúng em xin chân thành cảm ơn và mong nhận sự góp ý, chỉ bảo của thầy.

Sinh viên thực hiện:

Phan Tuấn Anh

Phạm Minh Trung

# 

# 

# 

# NHIỆM VỤ ĐỀ TÀI

* Tìm hiểu tổng quan bài toán tô màu đồ thị.
* Ứng dụng của bài toán tô màu đồ thị.
* Giới thiệu một số thuật toán tô màu đồ thị và cài đặt.
* Mô phỏng một số ví dụ dựa vào các thuật toán đã nêu bằng trình biên dịch Dev-C++.

# BẢNG PHÂN CÔNG CÔNG VIỆC

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Họ Tên SV | Nội Dung Công Việc | Đóng Góp |
| 1 | Phan Tuấn Anh | - Ứng dụng của bài toán tô màu  - Thuật toán tô màu theo bậc và cài đặt  - Giải thuật tham lam và cài đặt  - Lập báo cáo | 55% |
| 2 | Phạm Minh Trung | - Tìm hiểu tổng quan về đồ thị và bài toán tô màu  - Giải thuật Welch-Powell và cài đặt  - Slide thuyết trình | 45% |

**MỤC LỤC**

[LỜI CẢM ƠN](#_Toc454581233)

[NHIỆM VỤ ĐỀ TÀI](#_Toc454581234)

[BẢNG PHÂN CÔNG CÔNG VIỆC](#_Toc454581239)

[CHƯƠNG 1: GIỚI THIỆU BÀI TOÁN 1](#_Toc454581240)

[**1.1. Tổng quan về đồ thị** 1](#_Toc454581241)

[*1.1.1. Khái niệm đồ thị* 1](#_Toc454581242)

[*1.1.2. Các khái niệm khác* 1](#_Toc454581243)

[1.1.2.1. Đồ thị có hướng 1](#_Toc454581244)

[1.1.2.2. Đồ thị vô hướng 1](#_Toc454581245)

[1.1.2.3. Đồ thị đơn và đa đồ thị 2](#_Toc454581246)

[1.1.2.4. Đồ thị hỗn hợp 2](#_Toc454581247)

[**1.2. Bài toán tô màu đồ thị** 2](#_Toc454581248)

[*1.2.1. Định nghĩa* 2](#_Toc454581249)

[*1.2.2. Bài toán bốn màu* 3](#_Toc454581250)

[1.2.2.1. Giả thiết 3](#_Toc454581251)

[1.2.2.2. Định lý bốn màu 3](#_Toc454581252)

[*1.2.3. Bài toán tô màu đỉnh* 3](#_Toc454581253)

[*1.2.4. Bài toán tô màu cạnh* 3](#_Toc454581254)

[*1.2.5. Tô màu đồ thị phẳng* 3](#_Toc454581255)

[CHƯƠNG 2: ỨNG DỤNG 4](#_Toc454581256)

[**2.1. Bài toán điều khiển đèn hiệu nút giao thông** 4](#_Toc454581257)

[*2.1.1. Bài toán* 4](#_Toc454581258)

[*2.1.2.Cách giải* 4](#_Toc454581259)

[**2.2. Bài toán lập lịch thi** 5](#_Toc454581260)

[*2.2.1. Bài toán* 5](#_Toc454581261)

[*2.2.2.Cách giải* 5](#_Toc454581262)

[*2.2.3.Ví dụ* 5](#_Toc454581263)

[**2.3. Bài toán phân chia tần số** 6](#_Toc454581264)

[*2.3.1. Bài toán* 6](#_Toc454581265)

[*2.3.2. Cách giải* 6](#_Toc454581266)

[**2.4. Một số ứng dụng khác** 6](#_Toc454581267)

[CHƯƠNG 3: CÁC THUẬT TOÁN 6](#_Toc454581268)

[**3.1. Tô màu theo bậc** 7](#_Toc454581269)

[*3.1.1. Thuật giải:* 7](#_Toc454581270)

[*3.1.2. Ưu khuyết điểm* 7](#_Toc454581271)

[**3.2. Giải thuật tham lam(Greedy)** 7](#_Toc454581272)

[*3.2.1. Thuật giải* 7](#_Toc454581273)

[*3.2.2. Ưu khuyết điểm* 7](#_Toc454581274)

[**3.3. Thuật giải Welch -Powell** 8](#_Toc454581275)

[*3.3.1. Thuật giải* 8](#_Toc454581276)

[CHƯƠNG 4: CÀI ĐẶT THUẬT TOÁN 8](#_Toc454581277)

[**4.1. Tô màu theo bậc** 8](#_Toc454581278)

[*4.1.1. Cài đặt* 8](#_Toc454581279)

[*4.1.2. Hướng dẫn sử dụng* 10](#_Toc454581280)

[**4.2. Thuật giải Greedy** 12](#_Toc454581281)

[*4.2.1. Cài đặt* 12](#_Toc454581282)

[*4.2.2. Hướng dẫn sử dụng* 14](#_Toc454581283)

[**4.3 Thuật giải Welch-Powell** 17](#_Toc454581284)

[*4.3.1. Cài đặt* 17](#_Toc454581285)

[*4.3.2. Hướng dẫn sử dụng* 19](#_Toc454581286)

[CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 22](#_Toc454581287)

[**5.1. Kết luận** 22](#_Toc454581288)

[**5.2. Hướng phát triển** 22](#_Toc454581289)

[TÀI LIỆU THAM KHẢO 22](#_Toc454581290)

**NHẬN XÉT**

................................................................................................................................................................................................................................................................................................................................................................................................................................................ ................................................................................................................................................................................................................................................................................................ ................................................................................................................................................................................................................................................................................................ ................................................................................................................................................................................................................................................................................................ ................................................................................................................................................................................................................................................................................................ ................................................................................................................................................................................................................................................................................................ ................................................................................................................................................................................................................................................................................................ ................................................................................................................................................................................................................................................................................................ ................................................................................................................................................................................................................................................................................................ ................................................................................................................................................................................................................................................................................................ ................................................................................................................................................................................................................................................................................................ ................................................................................................................................................................................................................................................................................................ ................................................................................................................................................................................................................................................................................................

TP.HCM, Tháng 6, Năm 2016

GVHD

Trương Hải Bằng

# CHƯƠNG 1: GIỚI THIỆU BÀI TOÁN

## **1.1. Tổng quan về đồ thị**

### *1.1.1. Khái niệm đồ thị*

Trong toán học và tin học, đồ thị là đối tượng nghiên cứu cơ bản của lý thuyết đồ thị. Một cách không chính thức, đồ thị là một tập hợp các đối tượng được gọi là các đỉnh (hoặc nút) nối với nhau bởi các cạnh (hoặc cung). Cạnh có thể có hướng hoặc vô hướng. Đồ thị thường được vẽ dưới dạng một tập các điểm (các đỉnh nối với nhau bằng các đoạn thẳng (các cạnh).

Đồ thị biểu diễn được rất nhiều cấu trúc, nhiều bài toán thực tế có thể được biểu diễn bằng đồ thị.

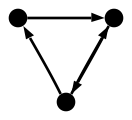
Cấu trúc đồ thị có thể được mở rộng bằng cách gán trọng số cho mỗi cạnh. Có thể sử dụng đồ thị có trọng số để biểu diễn nhiều khái niệm khác nhau.

### *1.1.2. Các khái niệm khác*

#### 1.1.2.1. Đồ thị có hướng

Đồ thị có hướng G là một cặp có thứ tự G=(V,A) trong đó:

* V, tập các đỉnh hoặc nút
* A, tập các cặp có thứ tự chứa các đỉnh, được gọi là các cạnh có hướng hoặc cung. Một cạnh e =(x,y) được coi là có hướng từ x tới y; x được gọi là điểm đầu/gốc và y được gọi là điểm cuối/ngọn của cạnh.

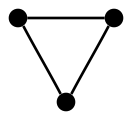


Ví dụ đồ thị có hướng

#### 1.1.2.2. Đồ thị vô hướng

Đồ thị vô hướng G là một cặp không có thứ tự G=(V,E) trong đó:

* V, tập các đỉnh hoặc nút
* E, tập các cặp không thứ tự chứa các đỉnh phân biệt, được gọi là cạnh. Hai đỉnh thuộc một cạnh được gọi là các đỉnh đầu cuối của cạnh đó.



Ví dụ đồ thị vô hướng

#### 1.1.2.3. Đồ thị đơn và đa đồ thị

Đồ thị đơn là đồ thị mà không có khuyên và không có cạnh song song. Đa đồ thị là đồ thị mà không thỏa đồ thị đơn.

#### 1.1.2.4. Đồ thị hỗn hợp

Đồ thị hỗn hợp G là một bộ ba có thứ tự G=(V,E,A) với V,E và A được định nghĩa như trên.

## **1.2. Bài toán tô màu đồ thị**

### *1.2.1. Định nghĩa*

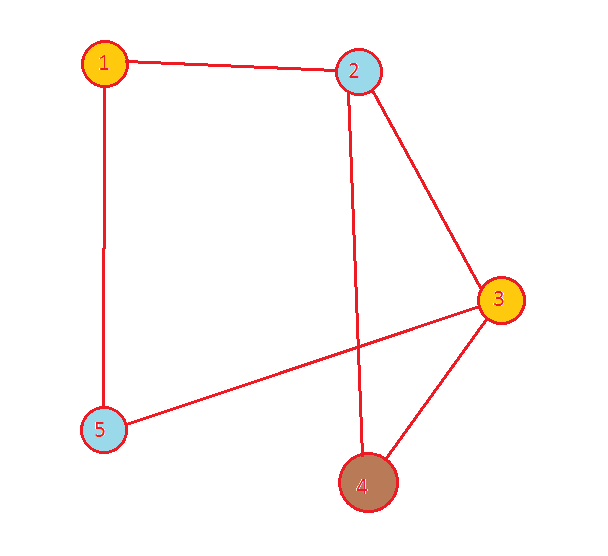
Phép tô màu các đỉnh đồ thị là phép gán màu cho các đỉnh của đồ thị sao cho hai đỉnh kề nhau có màu khác nhau.

Một cách hình thức có thể định nghĩa phép tô màu như sau:

Phép tô màu là một ánh xạ γ : X -> N sao cho:

٧(x,y) € X,γ(x)≠ γ(y)

Ví dụ:



Ví dụ tô màu đồ thị

Số màu ít nhất cần thiết dùng để tô màu các đỉnh cảu một đồ thị nào đó (giả sử là đồ ) được gọi là sắc số và được kí hiệu là γ(G).

### *1.2.2. Bài toán bốn màu*

#### 1.2.2.1. Giả thiết

Trên một bản đồ bất kì, ta nói nó được tô màu nếu mỗi miền của bản đồ được tô một màu xác định sao cho hai miền kề nhau (chung một phần biên) phải được tô màu khác nhau. Vấn đề đặt ra là phải dùng tối thiểu bao nhiêu màu để tô được một bản đồ bất kì . Vấn đề này được đưa ra vào năm 1982 do giáo sư De Morgan đặt ra: “Mọi bản đồ đều có thể tô bốn màu sao cho hai nước nằm kề nhau phải được tô bằng hai màu khác nhau ”. Sau đó có rất nhiều cố gắng của rất nhiều nhà toán học để giải bài toán này nhưng đều không đi đến kết quả cuối cùng. Cho đến năm 1976, một nhóm các nhà toán học (K.Appel, W.Heken, J.Koch) đã xây dựng một lời giải dựa trên kết quả do máy tính IBM cung cấp đã khẳng định giả thiết bốn màu là đúng.

#### 1.2.2.2. Định lý bốn màu

Mọi đồ thị phẳng có sắc số γ(G)≤4.

### *1.2.3. Bài toán tô màu đỉnh*

Tô màu đỉnh của một đồ thị là sự gán màu cho các đỉnh của nó một màu cụ thể sao cho không có 2 đỉnh kề nhau được gán cùng màu.

Một đồ thị có thể tô được bằng K màu, trong đó mỗi một tập các đỉnh cùng màu gọi là một lớp màu

Một đồ thị có thể được tô bằng K màu nghĩa là có có K tập độc lập trong đồ thị.

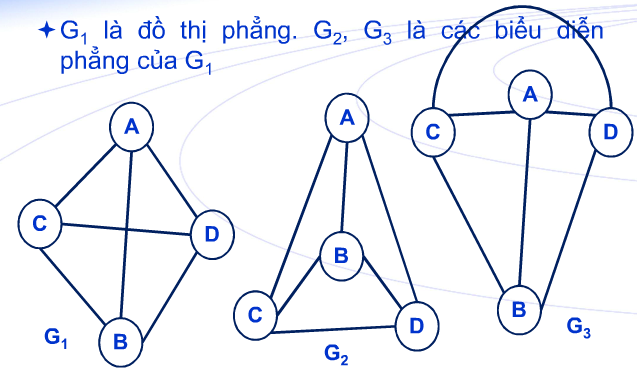
### *1.2.4. Bài toán tô màu cạnh*

Tô màu cạnh của một đơn đồ thị là sự gán màu cho các đỉnh của nó sao cho không có 2 cạnh kề được gán cùng một màu.

Đồ thị G được gọi là tô được bởi K màu cạnh nếu G có một phép tô màu K màu cạnh phù hợp. Thông thưởng hầu hết các đồ thị không là đồ thị khuyên đều tô được, và nếu G có tính chất như vậy thì G cũng có thể tô bởi 1 màu với 1>k.

### *1.2.5. Tô màu đồ thị phẳng*

Một đồ thị được gọi là phẳng nếu nó có thể vẽ được trên một mặt phẳng mà không có các cạnh nào cắt nhau (ở mọi điểm không phải là điểm mút của các cạnh), hình vẽ như thế gọi là một biểu diễn phẳng của đồ thị.



Ví dụ đồ thị phẳng

Mọi bản đồ tạo bởi các đưởng thẳng trên mặt phẳng có thể tô 2 màu.

Điều kiện cần và đủ để bản đồ có thể tô bằng 2 màu là mọi đỉnh của đồ thị phẳng tương ứng với bậc chẵn lớn hơn hoặc bằng 2.

# CHƯƠNG 2: ỨNG DỤNG

## **2.1. Bài toán điều khiển đèn hiệu nút giao thông**

### *2.1.1. Bài toán*

Giả sử ta cần thiết lập một quy trình điều khiển đèn hiệu ở nút giao thông phức tạp, nhiều giao lộ, sao cho trong một khoảng thời gian nhất định, một số tuyến được thông qua trong khi một số tuyến khác bị cấm để tránh xảy ra đụng độ.

Vấn đề đặt ra là phân hoạch các tuyến đường thành một số ít nhất các nhóm, sao cho các tuyến trong mỗi nhóm không đụng độ. Khi đó thời gian chờ đợi tối đa để được thông đường là ít nhất.

### *2.1.2.Cách giải*

Giả sử nút giao thông có n tuyến Ti, T2,..., Tn

Những tuyến giao nhau có thể dẫn đến đụng độ gọi là các tuyến xung khắc.

Như vậy đèn hiệu phải báo sao cho những tuyến xung khắc không đồng thời giao thông, và cho phép đồng thời lưu thông những tuyến không xung khắc.

Ta mô hình hóa bài toán bằng đồ thị và đưa về bài toán tô màu đồ thị như sau:

Các đỉnh của đồ thị là các tuyến đường, và hai tuyến kề nhau khi và chỉ khi chúng xung khắc.

Ta tô màu các đỉnh đồ thị sao cho các đỉnh kề nhau không cùng màu. Ta coi mỗi màu đại diện cho một pha điều khiển đèn báo: các tuyến cùng màu đó lưu thông. Như vậy bài toán ban đầu đưa về bài toán tô màu đồ thị với số màu ít nhất.

## **2.2. Bài toán lập lịch thi**

### *2.2.1. Bài toán*

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

### *2.2.2.Cách giải*

Để giải bài toán ta lập đồ thị có các đỉnh là các môn thi và hai môn thi kề nhau nếu có một học sinh thi cả hai môn này. Thời gian thi mỗi môn được biểu thị bằng các màu khác nhau. Như vậy bài toán lập lịch thi được đưa về bài toán tô màu đồ thị.

### *2.2.3.Ví dụ*

Có 7 môn thi với các thông tin như sau:

Môn 1: có các sinh viên A, B, C và D thi.

Môn 2: có các sinh viên A, E, F, G và H thi.

Môn 3: có các sinh viên B, E, I, J và K thi.

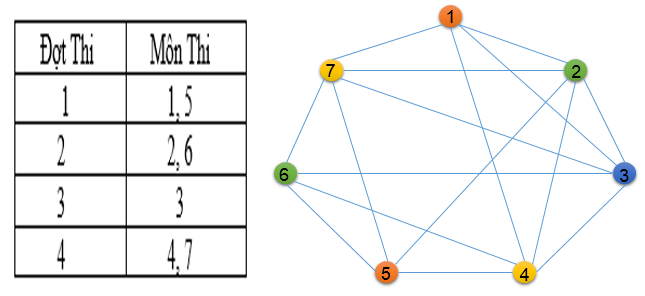
Môn 4: có các sinh viên B, F, L và M thi.

Môn 5: có các sinh viên G, L, N và O thi.

Môn 6: có các sinh viên J, M, N và P thi.

Môn 7: có các sinh viên D, H, K, O và P thi.

Giải:



## **2.3. Bài toán phân chia tần số**

### *2.3.1. Bài toán*

Có n đài phát. Hãy phân chia các kênh truyền hình cho các đài phát sao cho hai đài cách nhau không quá 100 (km) không được trùng kênh và số kênh dùng là ít nhất.

### *2.3.2. Cách giải*

Để giải bài toán ta lập đồ thị có các đỉnh là các đài phát và hai đài phát kề nhau nếu khoảng cách giữa chúng không quá 100 (km). Kênh truyền hình của mỗi đài được biểu thị bằng các màu khác nhau. Như vậy bài toán phân chia tần số được đưa về bài toán tô màu đồ thị.

## **2.4. Một số ứng dụng khác**

Ngoài một số ứng dụng bài toán tô màu đỉnh trên thì còn một số ứng dụng khác như bài toán thanh ghi, bản đồ địa dư, bài toán nữ sinh của Kirkman, bài toán chia thời khóa biểu...

# CHƯƠNG 3: CÁC THUẬT TOÁN

Có ba thuật toán chính thường được sử dụng vào bài toán tô màu:

* Tô màu theo bậc
* Tô màu dựa vào thuật giải tham lam (Greedy)
* Tô màu dựa vào giải thuật Welch-Powell

## **3.1. Tô màu theo bậc**

### *3.1.1. Thuật giải:*

Lặp lại các bước sau cho đến khi nào tô màu hết các đỉnh

* Bước 1: Chọn đỉnh có bậc lớn nhất tô màu i.
* Bước 2: Hạ bậc

- Đỉnh đã tô màu : bậc =0

- Những đỉnh có liên hệ : bậc = bậc -1

* Bước 3: Đánh dấu các đỉnh liên hệ (bậc vừa trừ đi 1) cấm tô màu i.

### *3.1.2. Ưu khuyết điểm*

Với thuật toán này chương trình sẽ chạy khá tốt, độ phức tạp của giải thuật là O(n2), tuy nhiên cho đến nay vẫn chưa có thuật toán nào thật sự tối ưu, không phải lúc nào việc tô màu cũng cho ra sắc số bé nhất.

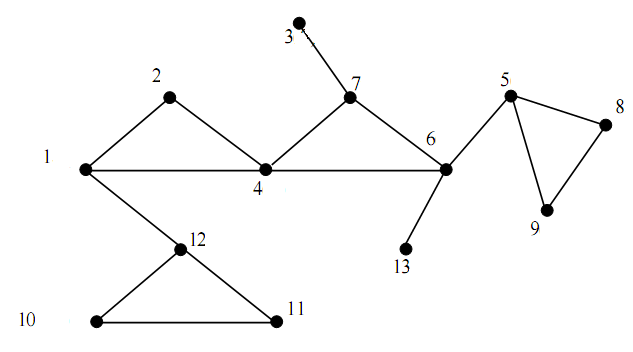
## **3.2. Giải thuật tham lam(Greedy)**

### *3.2.1. Thuật giải*

* Bước 1: i=0 (mọi đỉnh đều chưa được tô)
* Bước 2: i=i+1 . Tô màu i cho tất cả các đỉnh có thể tô được
* Bước 3: Lặp lại bước 2 cho đến khi tất cả các đỉnh đều được tô màu

### *3.2.2. Ưu khuyết điểm*

Với thuật toán này việc cài đặt trở nên dễ dàng hơn nhưng hiệu quả không cao, thuật toán chỉ đáp ứng được yêu càu bài toán đặt ra, nhưng không phải với số màu ít nhất có thể tô được, ví dụ như đồ thị bên dưới:



Ví dụ

Với đồ thị này áp dụng thuật giải trên ta cần phải dùng đến bốn màu trong khi chỉ cần đến ba màu là đủ.

## **3.3. Thuật giải Welch -Powell**

### *3.3.1. Thuật giải*

* Bước 1: i=0. Sắp xếp các đỉnh của G theo bậc giảm dần
* Bước 2: i=i+1. Dùng một màu để tô đỉnh đầu tiên và cũng dùng màu này để tô màu các đỉnh liên tiếp trong danh sách mà không kề với đỉnh đầu tiên.
* Bước 3: Bắt đầu trở lại danh sách, tô màu thứ hai cho đỉnh chưa được tô và lặp lại quá trình cho đến khi tất cả các đỉnh đều được tô màu .

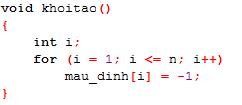
# CHƯƠNG 4: CÀI ĐẶT THUẬT TOÁN

## **4.1. Tô màu theo bậc**

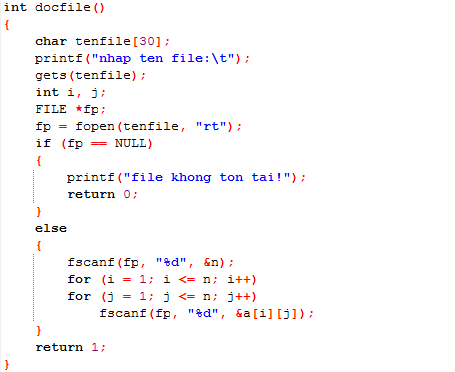
### *4.1.1. Cài đặt*

Mô tả:

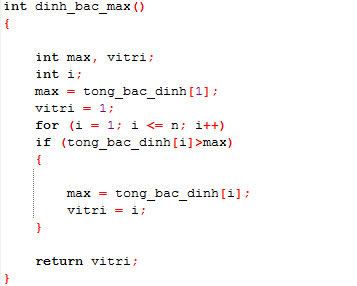
* Hàm khởi tạo (Lúc đầu các đỉnh chưa được tô)



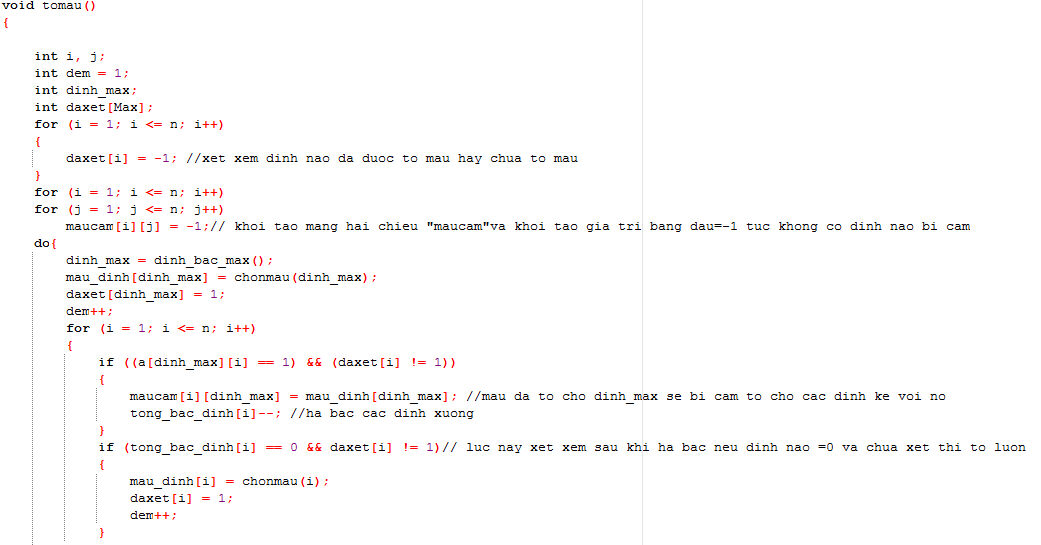
* Đọc dữ liệu từ file tạo sẵn.

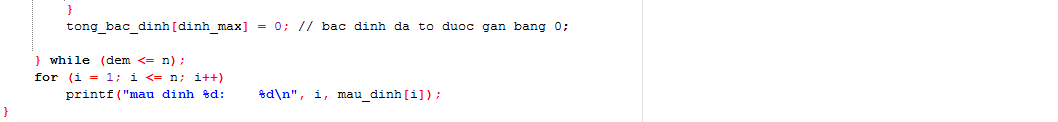


* Hàm tra kết quả vị trí của đỉnh có bậc cao nhất



* Hàm tô màu

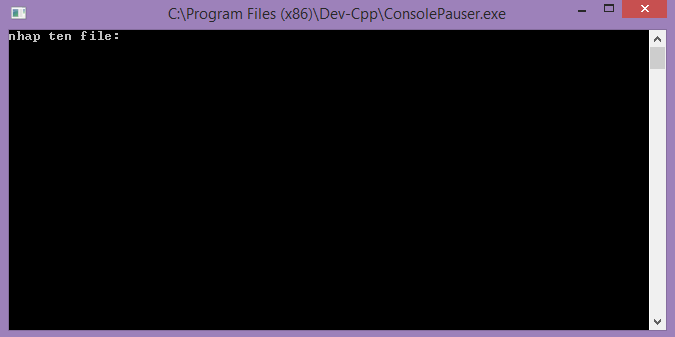




### *4.1.2. Hướng dẫn sử dụng*

Với cách cài đặt đầu tiên này, việc nhập dữ liệu được tách ra thành một khâu riêng bằng cách chúng ta phải tạo một file txt theo chuẩn đã trình bày ở trên. ( Trong bộ báo cáo có một bộ các file txt chính là ví dụ đi kèm với cách cài đặt này). Với cách nhập dữ liệu như thế này sẽ đơn giản hơn khi thực hiện chương trình. Lúc này việc nhập dữ liệu chỉ còn là nhập filename của file chứa ví dụ cần thực hiện và xem kết quả .

Nhập filename



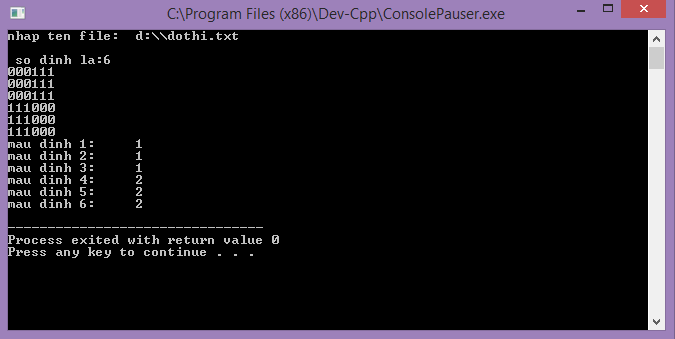
Sau đó ấn enter để xem nội dung file và kết quả

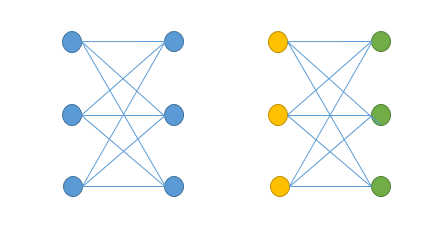
Kết quả:

Trong phần kết quả sẽ in ra:

* Nội dung của ma trận biểu diễn đồ thị
* Các đỉnh cụ thể được tô màu theo thứ tự

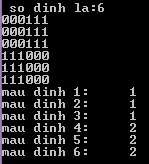
Kết quả của ví dụ trên:



Sau đây ta sẽ so sánh lại kết quả:

4

1



6

5

4

5

2

6

3

Một số bộ kiểm thử khác:

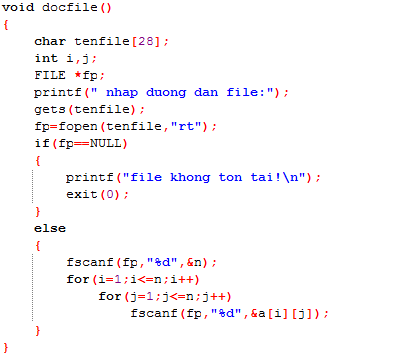
|  |  |
| --- | --- |
| Input | Ouput |
|  |  |
|  |  |

## **4.2. Thuật giải Greedy**

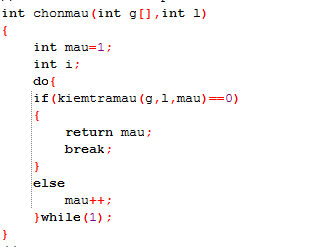
### *4.2.1. Cài đặt*

Mô tả:

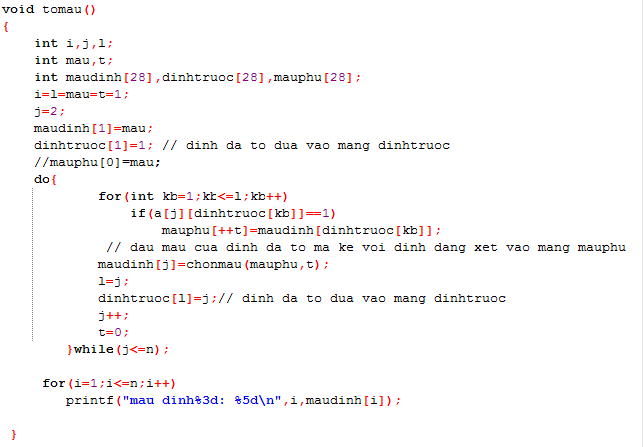
* Hàm đọc dữ liệu từ file tại sẵn



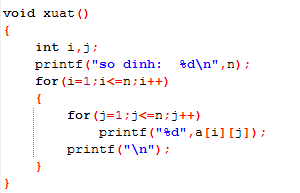
* Hàm chọn màu để tô



* Hàm tô màu cho các đỉnh đồ thị



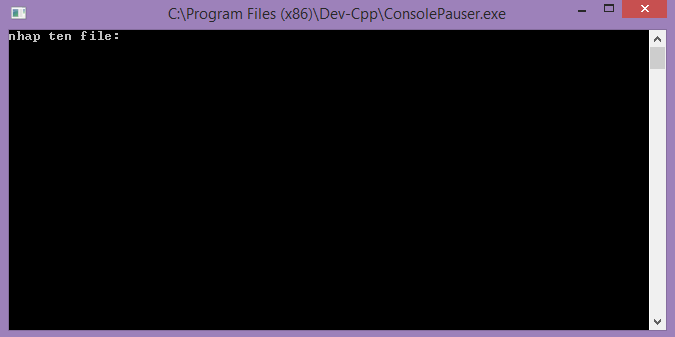
* Hàm xuất



### *4.2.2. Hướng dẫn sử dụng*

Tương tự thuật giải tô màu theo bậc, việc nhập dữ liệu được tách ra thành một khâu riêng bằng cách chúng ta phải tạo một file txt theo chuẩn đã trình bày ở trên.

Nhập filename

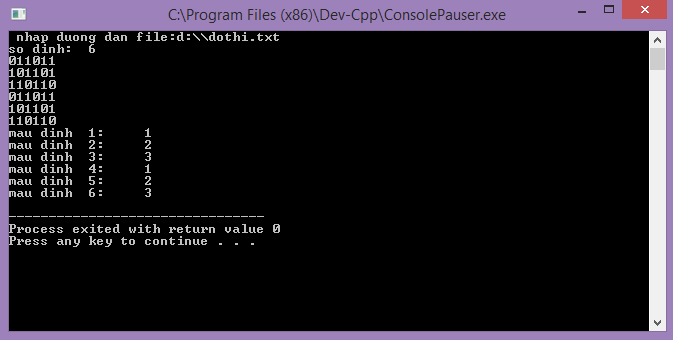


Sau đó ấn enter để xem nội dung file và kết quả

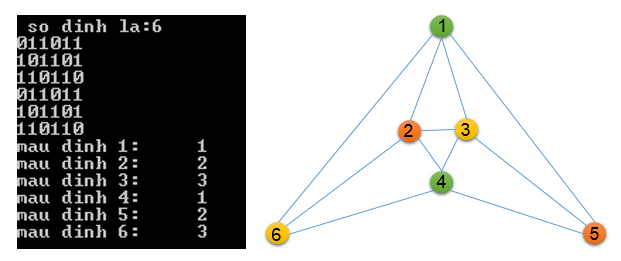
Trong phần kết quả sẽ in ra:

* Nội dung của ma trận biểu diễn đồ thị
* Các đỉnh cụ thể được tô màu theo thứ tự

Kết quả của ví dụ trên:



So sánh lại kết quả:



Một số bộ kiểm thử khác:

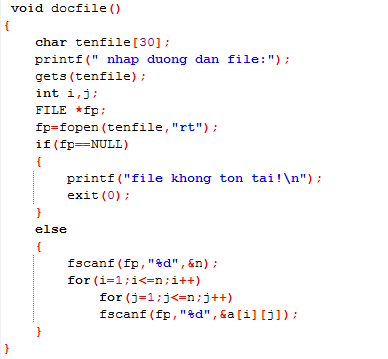
|  |  |
| --- | --- |
| Input | Ouput |
|  |  |
|  |  |

## **4.3 Thuật giải Welch-Powell**

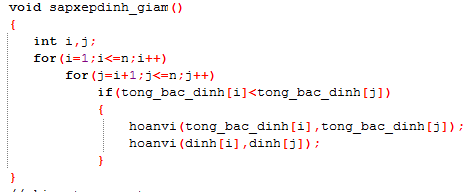
### *4.3.1. Cài đặt*

Mô tả:

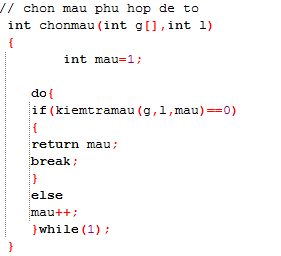
* Hàm đọc file



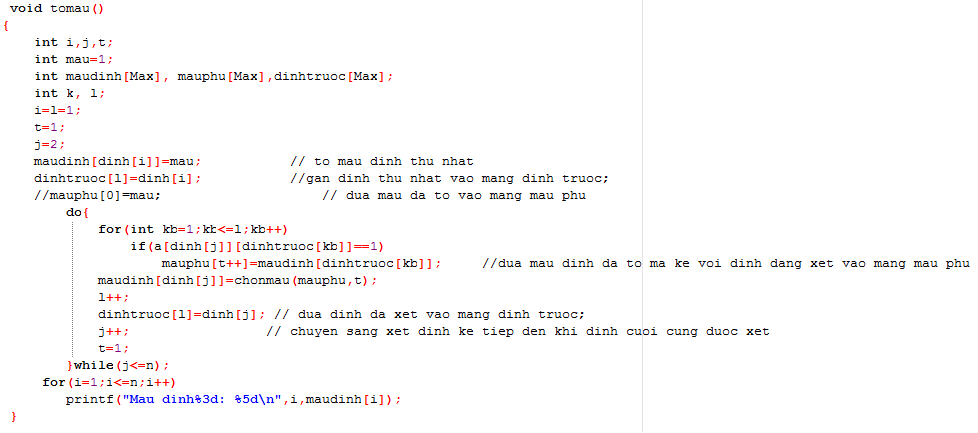
* Hàm sắp xếp đỉnh theo thứ tự giảm dần



* Hàm chọn màu



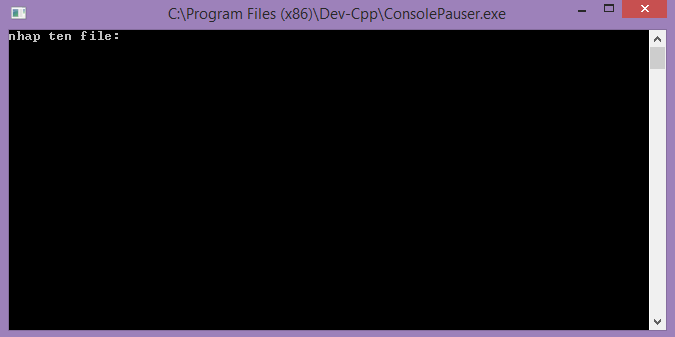
* Hàm tô màu



### *4.3.2. Hướng dẫn sử dụng*

Tương tự thuật giải tô màu theo bậc, việc nhập dữ liệu được tách ra thành một khâu riêng bằng cách chúng ta phải tạo một file txt theo chuẩn đã trình bày ở trên.

Nhập filename

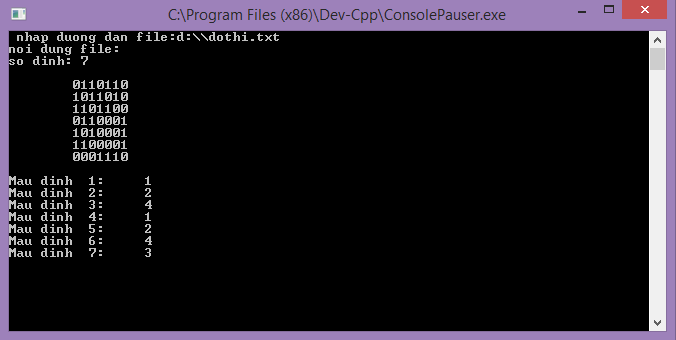


Sau đó ấn enter để xem nội dung file và kết quả

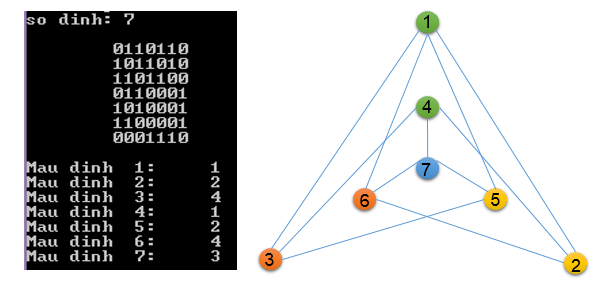
Trong phần kết quả sẽ in ra:

* Nội dung của ma trận biểu diễn đồ thị
* Các đỉnh cụ thể được tô màu theo thứ tự

Kết quả của ví dụ trên:



So sánh lại kết quả:



Một số bộ kiểm thử khác:

|  |  |
| --- | --- |
| Input | Ouput |
|  |  |
|  |  |

# CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## **5.1. Kết luận**

Qua tìm hiểu và nghiên cứu về các thuật toán tô màu bài toán đồ thị và ứng dụng, chúng ta thấy rằng nhứng bài toán này rất gần gũi với thực tế. Ứng dụng việc sử dụng các thuật toán trên giải quyết được các vấn đề trong thực tế được nghiên cứu và áp dụng vào thực tế như: việc sắp xếp lịch thi, thới khóa biểu trong trường học, bản đồ địa dư, đèn hiệu giao thông,... .

### **5.2. Hướng phát triển**

Các chương trình được thực hiện ở trên đang trong quá trình hoàn thiện, phát triển nâng cao. Trong thời gian ngắn nhất chúng em sẽ tìm cách giải bài toán một cách tốt nhất phát triển nó ở ngôn ngữ bậc cao như C#, Dot Net,... Tiến tới lập trình WinForm, thiết kế giao diện..., kết hợp các thuật toán khác ứng dụng vào thực tế. Từ đó, đánh giá giải thuật tìm ra bài toán tối ưu xây dựng các bài toán lớn.

# TÀI LIỆU THAM KHẢO

[1] Huỳnh Minh Trí – Lý Thuyết Đồ Thị, ĐH Sài Gòn, 2008

[2] Reinhard Diestel, Graph Theory, Springer-Verlag Heidelberg, NewYork 1997-2002

[3] Graph coloring problems and their applications in scheduling - Da’niel Mark

[4] http://myweb.pro.vn/tham-khao-tai-lieu/49205

[5] <http://en.wikipedia.org/wiki/Edge_coloring>

[6] <http://www.cs.sunysb.edu/~algorith/files/edge-coloring.shtml>

# 