

Bài tập lớn 3

Version: 1.00

I- Giới thiệu

Trong bài tập lớn 3 này, sinh viên sẽ lập trình cho một bài toán cụ thể. Các kiến thức sau đây sẽ được sử dụng trong bài tập lớn:

- Graph
- Thao tác đọc dữ liệu từ file text
- Tìm hiểu và hiện thực 1 giải thuật để giải quyết 1 bài toán cụ thể

Bài toán cho bài tập lớn này là bài toán “**Tô màu đồ thị**”.

II- Mô tả

1. Bài toán

Trong *Lý thuyết đồ thị*, **tô màu đồ thị** (tiếng Anh: *graph coloring*) là bài toán trong đó tập đỉnh hay tập cạnh hay tập miền của đồ thị có thể được gán bởi một tập các màu thỏa mãn ràng buộc nào đó. Tô màu đồ thị có thể là:

- **Tô màu đỉnh** (tiếng Anh: *vertex coloring*) là gán cho mỗi đỉnh của đồ thị một màu nào đó sao cho không có hai đỉnh liên kề nào có màu trùng nhau.
- **Tô màu cạnh** (tiếng Anh: *edge coloring*) là gán cho mỗi cạnh của đồ thị một màu nào đó sao cho không có 2 cạnh nào có đỉnh chung trùng màu.
- **Tô màu miền** (tiếng Anh: *face coloring*) là gán cho mỗi miền của đồ thị phẳng một màu sao cho không có 2 miền có chung đường biên lại cùng màu.

Sắc số (tiếng Anh: *chromatic number*) của một đồ thị là **số màu ít nhất** để tô các đỉnh.

Bài toán tô màu cạnh hay tô màu miền đều có thể chuyển về dạng tô màu đỉnh. Do đó trong bài tập lớn này chúng ta chỉ cần giải quyết bài toán tô màu đỉnh.

Rất nhiều bài toán trong thực tế có thể mô hình hóa thành bài toán tô màu đồ thị. VD: xếp lịch thi, xếp thời khóa biểu, xếp hội đồng chấm luận văn, xếp phòng, tô màu bản đồ thế giới.

2. Các bước hiện thực lời giải

Sinh viên có thể hiện thực bài toán tô màu theo các bước sau:

- 1 Đọc file input
- 2 Xây dựng cấu trúc Graph phù hợp
- 3 Dựa vào Graph vừa xây dựng tiến hành tô màu

3. Dữ liệu nhập

Sinh viên được cung cấp file *input.txt* gồm các thành phần sau:

- Dòng thứ 1 chứa 1 chữ số: số đỉnh của đồ thị.
- Dòng thứ 2 trở đi: chứa các kí tự chữ cái in hoa, chữ cái đầu tiên là tên của đỉnh đang xét, các chữ cái tiếp theo là tên của các đỉnh có cạnh nối với đỉnh đang xét.

VD:

4

A B C

B A

C A

D

Đồ thị trên gồm 4 đỉnh, đỉnh A có cạnh nối đến 2 đỉnh {B, C}, đỉnh B có cạnh nối đến đỉnh A, đỉnh C có cạnh nối đến đỉnh A, đỉnh D là đỉnh cô lập.

4. Hiện thực và biên dịch bài tập lớn

Sinh viên đọc kỹ đặc tả nội dung bài tập lớn, tìm hiểu các giải thuật tô màu đã có (nguồn Internet).

Trong bài tập lớn này, sinh viên được cung cấp sẵn một số files mã nguồn bao gồm:

- `defs.h`: file định nghĩa các cấu trúc dùng chung.
- `main.cpp`: chứa hàm `main()` sẽ được gọi lúc chương trình thực thi (sinh viên không được thay đổi nội dung của file này)
- **GraphColoring.cpp**: phân hiện thực giải thuật tô màu.
- `input.txt`: chứa 1 input mẫu.

Sinh viên chỉ chỉnh sửa một file duy nhất **GraphColoring.cpp** để định nghĩa và hiện thực các nội dung của bài tập lớn. Sinh viên có thể thêm, bớt các hàm, thủ tục hay biến, ... tuy nhiên sinh viên **không được phép sử dụng thêm thư viện nào**, chỉ sử dụng các thư viện đã được include sẵn. Phương thức `GraphColoring()` mặc định sẽ được gọi từ chương trình chính, vì vậy sinh viên **không được phép chỉnh sửa tên cũng như các tham số**.

Sinh viên biên dịch và thực thi theo cú pháp sau:

```
g++ main.cpp -o main.exe
```

```
./main.exe
```

5. Chấm bài tập lớn

Bài tập lớn sẽ được chấm dựa trên testcase, bài làm của sinh viên sẽ được biên dịch và thực thi trên số lượng lớn các testcase khác nhau. Một số lưu ý trong chấm bài tập lớn như sau:

- Sẽ chấm dựa vào “**sắc số**” giữa bài làm của sinh viên so với **solution**, điểm sẽ từ 0 đến 1.1 (nếu đáp án nhỏ hơn **solution** thì sẽ tính là 1.1). VD: sắc số do bài làm của sinh viên tính ra được là 4, sắc số solution là 3, điểm sẽ tính như sau: $1 - (4-3)/3.0$
- Để hạn chế việc sinh viên dùng phương pháp quét cạn để giải quyết bài toán này, thời gian thực thi cho mỗi testcase là 3s → 15s (tùy theo độ phức tạp của đồ thị).
- Số lượng testcases sẽ phân bổ như sau: 30% dễ, 30% trung bình, 30% khá khó, 10% khó.

III- Nộp bài

Khi nộp bài, sinh viên sử dụng account đã được cấp phát để nộp bài qua hệ thống Sakai. Sinh viên chỉ nộp đúng một file **GraphColoring.cpp** (tên file phải được viết đúng dạng chữ hoa, chữ thường). File được nộp phải là file chương trình gốc, SINH VIÊN **KHÔNG ĐƯỢC NÉN FILE** KHI NỘP BÀI. Sinh viên phải kiểm tra chương trình của mình trên Cygwin trước khi nộp.

Thời hạn chót để nộp bài là **11:55 AM thứ 2 ngày 25/11/2013**. **KHÔNG** nhận bài được gửi qua mail hoặc bất kỳ hình thức nào khác. Bài nộp trễ sẽ **KHÔNG** được nhận.

IV- Xử lý gian lận

Bài tập lớn phải được sinh viên **TỰ LÀM**. Sinh viên sẽ bị coi là gian lận nếu:

- Có sự giống nhau bất thường giữa mã nguồn của các bài nộp. Trong trường hợp này, **TẤT CẢ** các bài nộp đều bị coi là gian lận. Do vậy sinh viên phải bảo vệ mã nguồn bài tập lớn của mình.
- Sinh viên không hiểu mã nguồn do chính mình viết, trừ những phần mã được cung cấp sẵn. Sinh viên có thể tham khảo từ bất kỳ nguồn tài liệu nào, tuy nhiên phải đảm bảo rằng mình hiểu rõ ý nghĩa của tất cả những dòng lệnh mà mình viết. Trong trường hợp không hiểu rõ mã nguồn của nơi mình tham khảo, sinh viên được đặc biệt cảnh báo là **KHÔNG ĐƯỢC** sử dụng mã nguồn này; thay vào đó nên sử dụng những gì đã được học để viết chương trình.

Trong trường hợp bị kết luận là gian lận, sinh viên sẽ bị điểm 0 cho toàn bộ môn học (không chỉ bài tập lớn).

KHÔNG CHẤP NHẬN BẤT KỲ GIẢI THÍCH NÀO VÀ KHÔNG CÓ BẤT KỲ NGOẠI LỆ NÀO!

Sau mỗi bài tập lớn được nộp, sẽ có một số sinh viên có thể được gọi phỏng vấn ngẫu nhiên để chứng minh rằng bài tập lớn vừa được nộp là do chính mình làm.