

BÀI TẬP LỚN 1

MÔN PHÂN TÍCH VÀ THIẾT KẾ GIẢI THUẬT

I. Mục tiêu:

1. Rèn luyện kỹ năng lập trình và xây dựng ứng dụng, cụ thể là phần mềm xếp lịch thi.
2. Rèn luyện kỹ năng mô hình hóa bài toán thực tế và giải chúng thông qua bài toán mô hình, cụ thể là bài toán lập lịch (scheduling) và tô màu đồ thị (graph coloring).
3. Nghiên cứu lớp bài toán “khó” là NP-Đầy đủ và ứng dụng một vài giải thuật kinh nghiệm (heuristic) hoặc giải thuật ngẫu nhiên (randomized algorithm) cho lớp bài toán NP-Đầy đủ.

II. Đặc tả của bài tập

Trường Đại học Tôn Đức Thắng muốn xây dựng một phần mềm xếp lịch thi cho tất cả các sinh viên của trường. Các ràng buộc mà phần mềm cần đáp ứng gồm có **ràng buộc cứng (hard constraints)** và **ràng buộc mềm (soft constraints)**.

Ràng **buộc cứng** là những ràng buộc bắt buộc phần mềm phải đáp ứng như:

- Tất cả các môn học phải được xếp lịch thi và được cấp phòng thi tương ứng
- Sinh viên không thi hai môn cùng một thời điểm.
- Tất cả các nhóm cùng một môn học phải thi cùng ngày, cùng giờ để đảm bảo tính công bằng và tiết kiệm chi phí. Thông thường, một nhóm lý thuyết được chia làm hai hoặc ba tổ thực hành. Ví dụ, môn cấu trúc dữ liệu và giải thuật 1 có tất cả 04 nhóm lý thuyết, mỗi nhóm lý thuyết chia làm 2 tổ thực hành. Do đó, tổng số tổ thực hành của môn này là 08. Nếu đây là kỳ thi giữa kỳ thì cần 8 phòng máy (PM) khác nhau. Nếu đây là kỳ thi cuối kỳ thì chỉ cần 04 phòng thi lý thuyết (LT) khác nhau. Và, tất cả các tổ thi hoặc nhóm thi phải diễn ra cùng một thời điểm (trên nhiều phòng khác nhau).
- Sinh viên thi tối đa 3 môn cho mỗi ngày thi.

Ràng **buộc mềm** là những ràng buộc không thỏa cũng được nhưng nếu thỏa được thì phần mềm sẽ “đẹp và duyên dáng” hơn. Do đó, phần mềm thỏa được càng nhiều ràng buộc mềm càng tốt như:

- Tận dụng tối đa công suất của phòng thi.
- Lịch thi phải đảm bảo yếu tố công bằng giữa các thí sinh. Ví dụ, kỳ thi diễn ra trong 7 ngày. Sinh viên A có 4 môn thi. Sinh viên B có 4 môn thi. Lịch thi của sinh viên A chỉ tập trung vào 2 ngày đầu của kỳ thi (không có thời gian ôn bài). Lịch thi của sinh viên B trải dài ra 7 ngày (ngày nghỉ ngày thi nên có thêm thời gian ôn bài so với sinh viên A). Như vậy, lịch thi này không công bằng với sinh viên A.

III. Nhiệm vụ của bài tập

Xây dựng ứng dụng xếp lịch thi thỏa mãn:

1. Tất cả các ràng buộc cứng đã đặc tả ở trên
2. Việc quản lý được sự tồn tại nhiều nhóm của cùng một môn học. Một gợi ý về mặt cài đặt chương trình để đáp ứng tiêu chí này là mỗi đỉnh của đồ thị là một **class**. Trong mỗi **class**, chúng ta có thể bổ sung một thuộc tính có kiểu là danh sách để quản lý các nhóm của cùng môn học.
3. Việc xây dựng giao diện trực quan hỗ trợ nhiều góc nhìn: xem lịch thi theo môn, xem lịch thi theo tên SV, xem lịch thi theo ngày, xem lịch thi theo phòng,
4. Càng nhiều ràng buộc mềm càng tốt.

IV. Dữ liệu đầu vào và dữ liệu đầu ra của chương trình

Dữ liệu đầu vào là các file excel được mô tả như sau:

- Một file excel tên là **KetQuaDangKyMonHoc** mô tả kết quả đăng ký môn học của sinh viên gồm có các thuộc tính sau: mã môn học, tên môn học, mã nhóm, mã tổ, mã số sinh viên đã đăng ký thành công môn học đó, họ tên sinh viên, và email sinh viên. Trong đó, nếu mã tổ là rỗng thì đây chính là nhóm lý thuyết, nếu mã tổ là một số khác rỗng thì đây chính là tổ thực hành. Sinh viên xem file đính kèm để biết thêm chi tiết.
- Một file excel tên là **DanhSachPhongThi** mô tả các thông tin về phòng thi gồm có các thuộc tính: mã phòng, sức chứa, tính chất phòng (lý thuyết hay thực hành), và ghi chú (ghi nhận các loại phòng đặc biệt cho mỗi Khoa). Sinh viên xem file đính kèm để biết thêm chi tiết.

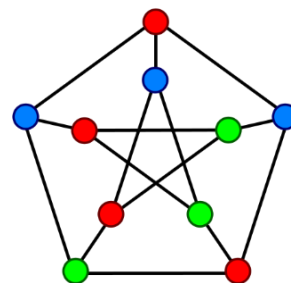
Dữ liệu đầu ra là kết quả xếp lịch thi cho từng sinh viên và lịch thi cho từng môn học. Thông tin lịch thi (có thể) là một file excel gồm có: mã môn, tên môn, mã nhóm, mã tổ, ngày thi, ca thi, phòng thi, sức chứa phòng thi, sĩ số sinh viên dự thi, loại phòng thi (Lý Thuyết - LT hay Phòng Máy – PM). Sinh viên xem thêm file **LichThi** để biết chi tiết. Ngoài ra, ứng dụng cần phải gán lịch thi cho từng SV ứng với toàn bộ các nhóm và các tổ của môn học. Sinh viên tự nghĩ ra các thuộc tính của file này để ghi ra kết quả.

V. Hướng dẫn mô hình bài toán

1. Bài toán tô màu đồ thị

Định nghĩa: một đồ thị có V đỉnh và E cạnh, ký hiệu là $G = \langle V, E \rangle$. Hai đỉnh có một cạnh nối gọi là hai đỉnh kề. Bài toán tô màu đồ thị yêu cầu tìm số màu ít nhất tô cho các đỉnh sao cho hai đỉnh kề phải khác màu.

Ví dụ: đồ thị như hình bên cần số màu tối thiểu là 3 (red, green, blue). Sinh viên đọc thêm về bài toán tại [ĐÂY](#).



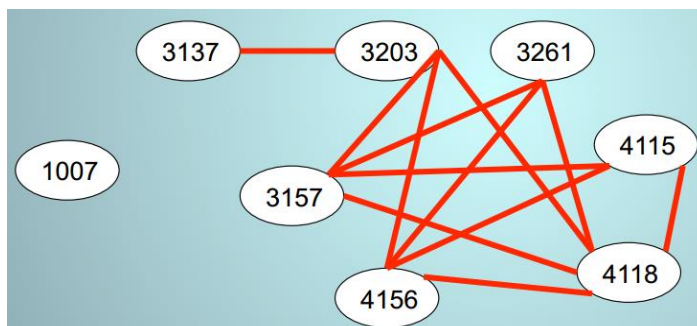
2. Quá trình mô hình bài toán xếp lịch thi thành bài toán tô màu đồ thị

- Xem mỗi môn học là một đỉnh của đồ thị.
- Nếu có một sinh viên đăng ký hai môn học thì giữa hai đỉnh tương ứng với hai môn đó tạo thành một cạnh (cạnh vô hướng). Nếu sinh viên đăng ký n môn học thì n đỉnh của đồ thị tương ứng với n môn đó tạo thành đồ thị đầy đủ có n đỉnh. Số màu tối thiểu cần dùng chính là số ca thi. Đồ thị này có trọng số chính là số lượng sinh viên học cả hai môn.
- Bài toán xếp lịch thi là trở thành bài toán tô màu đồ thị: gán các ca thi vào các môn học sao cho không có sinh viên nào thi hai môn cùng một thời điểm. Nghĩa là, hai đỉnh kề nhau phải khác màu.

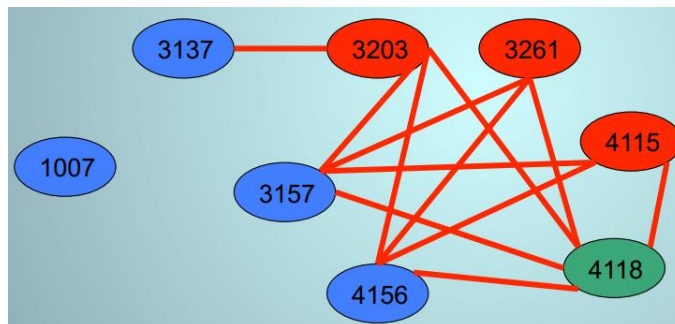
Ví dụ: Với kết quả đăng ký môn học của một vài sinh viên như sau:

Mã môn SV	1007	3137	3157	3203	3261	4115	4118	4156
51303179	x							
51301298		x		x				
51300988			x	x			x	
51201207				x			x	x
51203006						x	x	x
51201133					x		x	x
51201098			x			x	x	
51201456			x		x		x	

Giả sử thêm rằng số lượng giám thị và số lượng phòng thi đủ phục vụ kỳ thi. Với dữ liệu như trên thì đồ thị được xây dựng có hình minh họa như hình bên.



Như vậy, bài toán xếp lịch thi đã được chuyển thành bài toán tô màu đồ thị. Vấn đề còn lại là tìm thuật toán hiệu quả để tô màu đồ thị. Và với đồ thị như trên thì có thể tồn tại phương án tô màu như hình bên dưới:



Trong đó, mỗi màu là một ca thi. Như vậy, các môn có mã số 1007, 3137, 3157, 4156 có thể thi cùng một thời điểm. Kỳ thi này cần 3 ca thi.

Sinh viên đọc hiểu ví dụ và thử đưa các ràng buộc mềm vào bài toán và tự mô hình chúng trên đồ thị. Ví dụ: một thí sinh không thi quá 3 môn/ngày, không nghỉ quá 3 ngày giữa hai môn thi, các nhóm cùng môn nên thi chung một ngày....

3. Hướng dẫn nộp bài

Sản phẩm là một phần mềm đáp ứng đủ các tiêu chí và hạn cuối như sau:

Deadline 1 (05.10.2017) (3 điểm): hoàn thành chức năng đọc/ghi file và xây dựng thành công đồ thị mô hình, bài nộp là mã nguồn chương trình.

Deadline 2 (16.10.2017) (4 điểm): hoàn thành xếp lịch thi có **gán phòng thi** cho từng môn thi, bài nộp là mã nguồn chương trình và file lịch thi (dạng excel hoặc khác).

Deadline 3 (09.11.2017) (3 điểm): phần mềm hoàn thiện có giao diện trực quan, bài nộp là toàn bộ mã nguồn chương trình và file lịch thi xếp được.

Lưu ý:

- Sinh viên có thể chọn ngôn ngữ lập trình Java, C# hoặc C++ để xây dựng phần mềm xếp lịch thi.
- Yêu cầu của mỗi deadline là yêu cầu tối thiểu, nếu sinh viên làm vượt yêu cầu thì các deadline sau có thể nộp lại cùng một bản của deadline trước.
- Sau mỗi deadline mà sinh viên không nộp bài thì được 0 điểm bài tập lớn đó.

VI. Xử lý đạo văn:

Khoa xem xét và giải quyết vấn đề đạo văn như đã áp dụng trong các môn học trước. Nếu sinh viên được kết luận là đạo văn thì:

- Nhẹ nhất: được 0 điểm bài tập lớn và được đình chỉ thi cuối kỳ.
- Nặng hơn: được đề xuất đình chỉ học 1 năm vì vi phạm 3 nội dung đạo đức.

---HẾT---