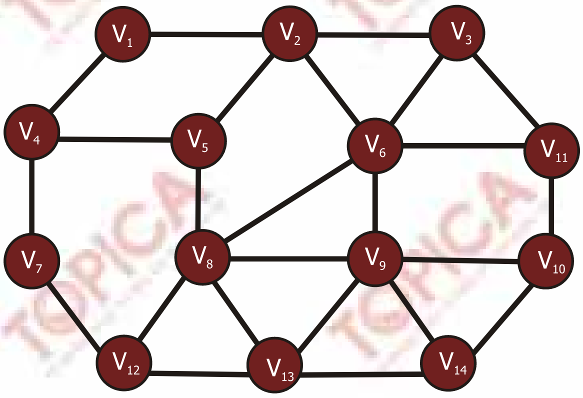
BÀI 5: NHỮNG KHÁI NIỆM CƠ BẢN CỦA LÝ THUYẾT ĐỒ THỊ



Giới thiệu

Lý thuyết đồ thị được xem như một lĩnh vực của toán học rời rạc, được phát triển từ lâu nhưng có nhiều ứng dụng hiện đại.

Vì những nội dung phong phú, xuất hiện trong nhiều lĩnh vực, lý thuyết đồ thị thường được trình bày như một giáo trình riêng với thời lượng có thể lên đến hàng trăm tiết. Nội dung bài học này chỉ đề cập những khái niệm và những ứng dụng dễ thấy nhất của lý thuyết đồ thị.

Nội dung

- Các định nghĩa và ví dụ về đồ thị
- Một số khái niệm và thuật ngữ
- Biểu diễn đồ thị trên máy tính
- Một số đồ thị đặc biệt

Thời lượng học

- 10 tiết

Mục tiêu

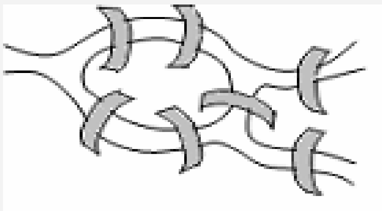
Sau khi học bài này, các bạn có thể:

- Nắm được các định nghĩa và cho được các ví dụ về đồ thị.
- Phân biệt được các loại đồ thị: đồ thị vô hướng, đồ thị có hướng, đơn đồ thị, đa đồ thị, đồ thị có trọng số.
- Trình bày được một số khái niệm và kể tên được một số thuật ngữ về đồ thị như: kề, bậc của đỉnh, đường đi, chu trình, liên thông, …
- Biết cách biểu diễn đồ thị trên máy tính.
- Minh họa một số khái niệm, kết quả và ứng dụng của lý thuyết đồ thị qua việc nghiên cứu một số đồ thị đặc biệt.

TÌNH HUỐNG DẪN NHẬP

Tình huống

Nội dung bài toán như sau: "Thành phố Konigberg được chia thành bốn vùng bởi các nhánh sông Pregel. Người ta xây 7 cây cầu nối các vùng này với nhau để làm một công viên như hình vẽ:



Câu hỏi

Có thể xuất phát tại một nơi nào đó trong thành phố, đi qua tất cả các cầu, mỗi cầu đúng một lần rồi lại trở về điểm xuất phát được không?

Lý thuyết đồ thị được xem như một lĩnh vực của toán học rời rạc, được phát triển từ lâu nhưng có nhiều ứng dụng hiện đại.

Một trong những kết quả đầu tiên của lý thuyết đồ thị có thể xem là lời giải của bài toán về 7 cây cầu ở Konigberg do Leonhard Euler đề xuất vào năm 1736. Tiếp theo vào năm 1845, Gustav Kirchhoff, bằng cách biểu diễn mạch điện như một đồ thị, đã tìm ra một định luật cho phép tính điện thế và cường độ dòng điện của mạch. Vào năm 1852, Fracis Guthrie đưa ra bài toán 4 màu, một bài toán nổi tiếng mà quá trình tìm lời giải của nó được xem như khai sinh ra lý thuyết đồ thị. Bài toán này mãi đến năm 1976 mới được giải quyết với sự trợ giúp của máy tính điện tử bởi các nhà toán học Mỹ là Kenneth Appel và Wolfgang Haken. Năm 1857, nhà hóa học Cayley đã tìm được công thức đếm số đồng đẳng của hydro cacbon no C_nH_{2n+2} bằng cách biểu diễn các dạng của công thức này như những cây, một lớp quan trọng của đồ thị có rất nhiều ứng dụng.

Vì những nội dung phong phú, xuất hiện trong nhiều lĩnh vực, lý thuyết đồ thị thường được trình bày như một giáo trình riêng với thời lượng có thể lên đến hàng trăm tiết. Với vài bài giảng trong giáo trình này, chúng tôi cũng chỉ đề cập được những khái niệm và những ứng dụng dễ thấy nhất của lý thuyết đồ thị. Những bạn đọc quan tâm nhiều hơn đến lý thuyết đồ thị có thể tham khảo những tài liệu chuyên sâu hơn về lý thuyết này.

5.1. Các định nghĩa và ví dụ về đồ thị

Định nghĩa: Đồ thị (graph) là một mô hình xây dựng trên một tập hữu hạn các đối tượng và những mối quan hệ hai ngôi trên chúng.

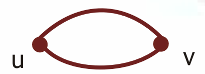
Một đồ thị G xem như một hệ thống gồm hai tập hợp (hữu hạn) mà ta ký hiệu cho gọn là G = (V, E), trong đó các phần tử của V được gọi là *đỉnh* (vertex) và các phần tử của E được gọi là *cạnh* (edge). Mỗi cạnh e của đồ thị được xem như hình thành bởi việc *liên kết* hai đỉnh u và v của đồ thị, và thường viết e = (u, v). Các đỉnh u, v được gọi là các *đỉnh mút* của e và e được gọi là *cạnh nối* u với v.

Để trực giác, đồ thị thường được biểu diễn trên mặt phẳng, trong đó mỗi đỉnh được vẽ như một điểm và mỗi cạnh được vẽ như một cung nối hai điểm. Trong trường hợp tổng quát nhất của mối quan hệ giữa đỉnh và cạnh, có thể xảy ra các tình huống:

- Giữa hai đỉnh có thể không có cạnh nối nào:

- Giữa hai đỉnh có thể có đúng một cạnh nối:

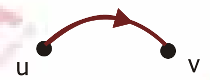
- Giữa hai đỉnh có thể có nhiều hơn một cạnh nối:



Như thế, nếu mỗi đỉnh của đồ thị mô tả một đối tượng, thì mỗi cạnh biểu diễn một quan hệ hai ngôi nào đó giữa các đối tượng này. Thông thường, vai trò của hai đỉnh mút u, v của cạnh e là như nhau, nghĩa là trong biểu diễn e = (u, v), thứ tự của u và v là không quan trọng. Trường hợp này, cạnh e mô tả một quan hệ đối xứng giữa u và v và e được gọi là *cạnh vô hướng*.

Trong một số ứng dụng, người ta đưa thêm hướng vào các cạnh để mô tả những quan hệ không đối xứng. Khi đó cặp đỉnh (u, v) biểu diễn cạnh e cần phải kể thứ tự và e được gọi là *cạnh có hướng*, đỉnh u được gọi là *đỉnh đầu* và đỉnh v được gọi là *đỉnh cuối* của cạnh e.

Để diễn đạt e = (u, v) là cạnh vô hướng, người ta thường nói e là *cạnh nối giữa* u và v, còn để diễn đạt e = (u, v) là cạnh có hướng, người ta thường nói e là *cạnh nối từ* u đến v và trên cung biểu diễn nó, người ta thêm vào chiều mũi tên hướng từ u đến v.



Ngoài ra trong định nghĩa cạnh, người ta cũng cho phép trường hợp cạnh nối một đỉnh với chính nó, nghĩa là e = (u, u). Một cạnh như vậy được gọi là *khuyên*. Một đỉnh có khuyên mô tả một phần tử có quan hệ đang xét với chính nó. 

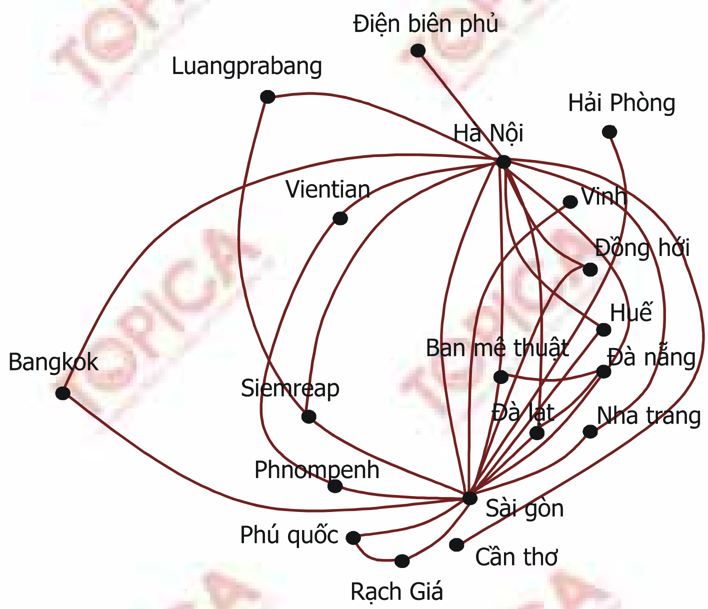
Để thuận tiện cho việc nghiên cứu, người ta phân loại đồ thị theo những định nghĩa hẹp hơn với những tên dành riêng. Dưới đây là những phân loại thông thường nhất.

- **Đồ thị vô hướng và đồ thị có hướng.** Một đồ thị mà tất cả các cạnh của nó đều vô hướng được gọi là *đồ thị vô hướng*, trái lại đồ thị được gọi là *có hướng*. Trong nhiều tình huống, một đồ thị vô hướng được xử lý như một đồ thị có hướng bằng cách đồng nhất một cạnh vô hướng bằng hai cạnh có hướng ngược chiều nhau. Ngược lại, nếu bỏ đi các hướng trên các cạnh, thì đồ thị có hướng trở thành đồ thị vô hướng.
- **Đơn đồ thị và đa đồ thị.** Một đồ thị không có khuyên và giữa hai đỉnh chỉ có nhiều nhất là một cạnh nối được gọi là một *đơn đồ thị*. Các đồ thị có khuyên hay có nhiều cạnh nối giữa hai đỉnh được gọi chung là các *đa đồ thị*. Như thế đơn đồ thị là một trường hợp riêng, nhằm nghiên cứu các mô hình mà người ta chỉ quan tâm việc giữa hai đối tượng khác nhau có hay không có mối quan hệ được xét.

  Một số khái niệm và kết quả nghiên cứu được phát biểu chung cho một số loại đồ thị, nhưng có những khái niệm và kết quả nghiên cứu được phát biểu riêng. Tùy từng tình huống mà trong việc trình bày sẽ nói rõ.

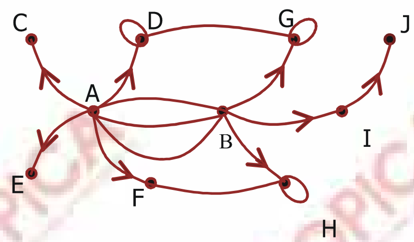
  Ví dụ về đồ thị rất phong phú, bao trùm trên nhiều lĩnh vực, từ khoa học tự nhiên đến khoa học xã hội. Dưới đây là một vài Ví dụ cho thấy sự đa dạng của đồ thị.
- **Mạng giao thông.** Một mạng giao thông (đường bộ, đường thủy, đường sắt, đường không) là những Ví dụ điển hình về đồ thị, trong đó mỗi nút giao thông là một đỉnh, đường nối từ nút này đến nút kia là một cạnh. Tùy tình huống mà mạng đang xét là đơn hay đa đồ thị, hoặc có hướng hay vô hướng. Nhiều thuật ngữ giao thông được dùng trong đồ thị như đường đi, chu trình, liên thông, cầu, lát cắt, luồng, ...

  Chẳng hạn trong một mạng hàng không của một hãng nào đó, mỗi sân bay được xem như một đỉnh, mỗi tuyến bay từ sân bay này đến sân bay kia được xem như một cạnh. Thông thường giữa hai sân bay có thể có hay không một tuyến bay và các tuyến bay này đều hai chiều (chú ý phân biệt tuyến bay với chuyến bay, trên một tuyến, có thể có nhiều chuyến bay khác nhau). Khi đó ta nhận được một đơn đồ thị vô hướng biểu diễn mạng hàng không đang xét.



Đơn đồ thị vô hướng biểu diễn mạng Vietnam Airlines

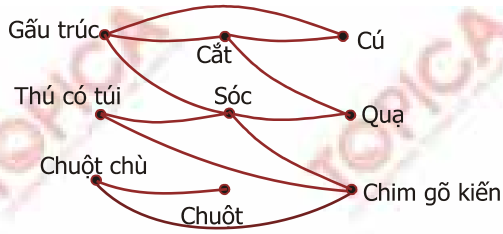
- **Mạng máy tính.** Một mạng máy tính bao gồm nhiều trung tâm máy tính nối lại với nhau để trao đổi thông tin và chia sẻ dữ liệu. Mỗi trung tâm như một đỉnh, đường nối từ trung tâm này đến trung tâm kia (gọi là kênh thoại) là một cạnh. Giữa hai trung tâm có thể không có, có một, hoặc có nhiều kênh thoại, có kênh thoại cho phép hai chiều, có kênh thoại chỉ được phép một chiều, có kênh thoại nối một trung tâm với chính nó (kênh nội bộ, dùng để thông báo chẳng hạn). Như vậy, ta nhận được một đa đồ thị có hướng biểu diễn mạng máy tính đang xét. Những kênh thoại hai chiều là những cạnh vô hướng, những kênh thoại một chiều là những cạnh có hướng. Một kênh thoại hai chiều có thể xem như hai kênh thoại một chiều có hướng ngược nhau.



Đa đồ thị có hướng biểu diễn một mạng máy tính

Trong mạng máy tính biểu diễn bằng đồ thị trên, có hai trung tâm A, B mà giữa chúng có 3 kênh thoại hai chiều để gửi tin tức cho nhau, ngoài ra chúng chỉ phát tin cho các trung tâm khác. Các trung tâm D, G, H là những trung tâm có kênh nội bộ.

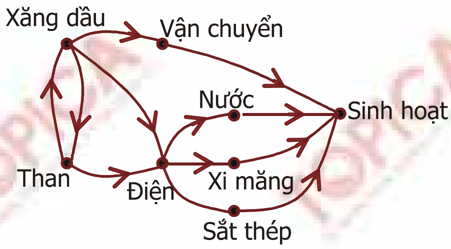
- **Đồ thị cạnh tranh.** Trong sinh vật, người ta quan tâm đến mối quan hệ cạnh tranh về nguồn thức ăn giữa các loài. Mối quan hệ này có thể mô tả bằng một đơn đồ thị vô hướng, trong đó mỗi đỉnh là một loài vật, hai loài vật nào có cạnh tranh về nguồn thức ăn được nối với nhau bằng một cạnh.



Đơn đồ thị mô tả mối quan hệ cạnh tranh giữa một số loài

Trong đồ thị trên, sóc cạnh tranh với bốn loài (gấu trúc, quạ, thú có túi, chim gõ kiến), trong khi chuột chỉ cạnh tranh với một loài (chuột chù).

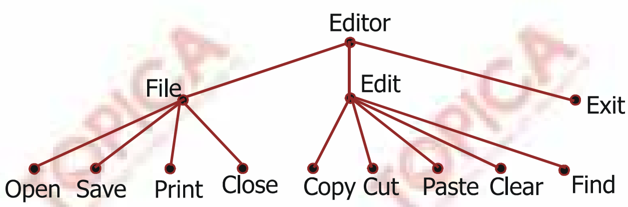
- **Đồ thị ảnh hưởng.** Đồ thị ảnh hưởng là một đơn đồ thị có hướng, mô tả mối quan hệ ảnh hưởng lẫn nhau giữa các đối tượng. Mỗi đối tượng được mô tả bằng một đỉnh, đỉnh u được nối một cạnh có hướng đến đỉnh v nếu u ảnh hưởng đến v. Chẳng hạn quan hệ về giá cả giữa các mặt hàng được mô tả trên đơn đồ thị có hướng dưới đây:



Đơn đồ thị có hướng mô tả ảnh hưởng giá cả

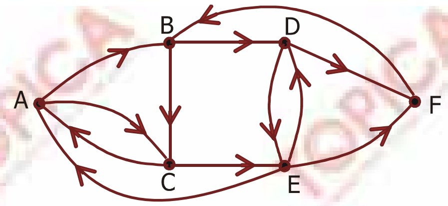
Đồ thị trên cho thấy giá các nhiên liệu xăng dầu và than ảnh hưởng qua lại lẫn nhau và ảnh hưởng đến giá các mặt hàng khác. Giá sinh hoạt chịu ảnh hưởng của tất cả các giá của các mặt hàng này.

- **Sơ đồ chức năng.** Là sơ đồ mô tả mối quan hệ giữa các chức năng của một tổ chức (nhà nước, tập đoàn, đảng phái, ...) hoặc của một phần mềm ứng dụng trên máy tính (soạn thảo văn bản, bảng tính, chỉnh sửa ảnh, ...). Mỗi chức năng được biểu diễn như một đỉnh và quan hệ giữa hai chức năng được biểu diễn như một cạnh. Đồ thị dưới đây biểu diễn các chức năng cơ bản của một hệ soạn thảo văn bản:



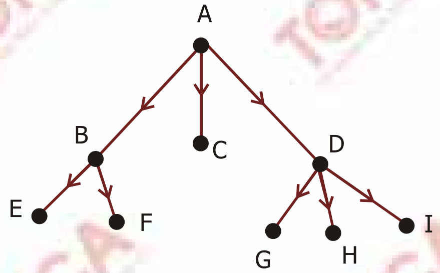
Đồ thị biểu diễn các chức năng soạn thảo văn bản

- **Sơ đồ liên kết.** Trên internet, mỗi trang web cung cấp một số thông tin cho người dùng, ngoài ra nó còn cho phép kết nối đến một số trang web nào đó. Trên các trang web hình thành một mối liên kết, trong đó từ một trang web, ta có thể đi đến những trang web khác. Một sơ đồ liên kết giữa các trang web như vậy có thể biểu diễn bằng một đồ thị có hướng, trong đó mỗi đỉnh là một trang web, còn mỗi cạnh hướng từ đỉnh A đến đỉnh B mô tả từ trang A ta đến được trang B.



Đồ thị liên kết của các trang web

- **Bản đồ gia phả.** Một bản đồ gia phả (phả hệ) cho thấy mối quan hệ huyết thống (trực hệ) trong một dòng họ, nó có thể được biểu diễn như một đơn đồ thị có hướng, trong đó mỗi thành viên trong họ (là đàn ông) được biểu diễn bằng một đỉnh. Đỉnh A có cạnh nối tới đỉnh B nếu A là bố của B.



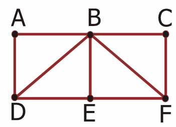
Đồ thị gia phả

Đồ thị gia phả trên mô tả một phả hệ gồm 3 đời, trong đó A là bố của B, C, D và là ông nội của E, F, G, H, I.

Từ một đồ thị, ta có thể trích ra các bộ phận khác nhau của chúng. Điều này dẫn đến định nghĩa sau:

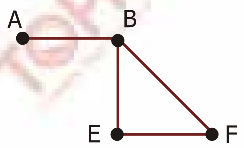
- **Đồ thị con.** *Đồ thị con* của một đồ thị G là một đồ thị nhận được từ G bằng cách bỏ đi một số cạnh và một số đỉnh (cùng với các cạnh liên quan đến những đỉnh này). Nói cách khác G’ = (V’, E’) là đồ thị con của G = (V, E) nếu $V' \subseteq V, E' \subseteq E$.

Ví dụ:

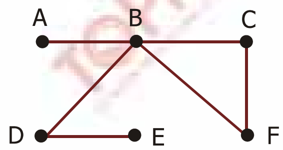


Đồ thị G

có các đồ thị con G_1 và G_2



Đồ thị G_1



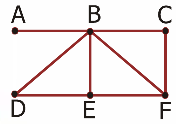
Đồ thị G_2

trong đó G_1 nhận được từ G bằng cách bỏ đi các đỉnh C, D và các cạnh nối đến C, D; còn G_2 nhận được từ G bằng cách bỏ đi các cạnh AD, BE và EF (các đỉnh giữ nguyên).

Ngược lại, từ một số đồ thị, ta có thể hợp chúng lại để nhận được một đồ thị mới theo định nghĩa sau:

Hợp hai đồ thị. *Hợp* của hai đơn đồ thị $G_1 = (V_1, E_1)$ và $G_2 = (V_2, E_2)$ là một đơn đồ thị có tập đỉnh là $V_1 \cup V_2$ và tập cạnh là $E_1 \cup E_2$. Hợp của hai đồ thị G_1 và G_2 được ký hiệu giống như hợp của hai tập hợp là $G_1 \cup G_2$.

Ví dụ: Hợp của hai đồ thị G_1 và G_2 trong Ví dụ trên cho ta đồ thị $G_1 \cup G_2$



5.2. Một số khái niệm và thuật ngữ

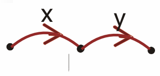
5.2.1. Kề

Đỉnh v được gọi là *kề với* đỉnh u nếu có cạnh hướng từ u đến v. Như vậy định nghĩa này bao gồm cả hướng. Trong trường hợp không xét hướng, hai đỉnh u, v được gọi là *kề nhau*.

Đỉnh v kề với đỉnh u

Hai đỉnh u, v kề nhau

Khái niệm kề cũng được phát biểu cho các cạnh. Cạnh y được gọi là *kề với* cạnh x nếu đỉnh cuối của x là đỉnh đầu của y. Nếu không kể hướng, hai cạnh x và y được gọi là *kề nhau* (nghĩa là x, y có một đỉnh chung).

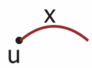


Cạnh y kề với cạnh x

Hai cạnh x, y kề nhau

5.2.2. Liên thuộc

Đỉnh u được gọi là *liên thuộc với* cạnh x nếu u là một đỉnh nút của x. Định nghĩa này không kể hướng của x là đi vào hay ra khỏi u, vì thế thường nói gọn là đỉnh u và cạnh x là *liên thuộc nhau*.



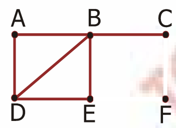
Đỉnh u và cạnh x liên thuộc nhau

5.2.3. Bậc của đỉnh

Khái niệm bậc của đỉnh được phát biểu riêng cho đồ thị vô hướng và đồ thị có hướng.

- **Bậc của đỉnh trong đồ thị vô hướng.** Ta gọi *bậc của đỉnh* u là số cạnh liên thuộc với nó. Bậc của u được ký hiệu là deg(u).

  Như thế bậc của đỉnh u đo mức độ quan hệ của đỉnh này đối với những đỉnh khác, bậc của đỉnh càng lớn thì mức độ quan hệ của đỉnh càng nhiều. Tên gọi ngã ba, ngã tư trong hệ thống giao thông chính là khái niệm bậc của đỉnh. Đỉnh bậc 0 còn được gọi là *đỉnh cô lập*, đỉnh bậc một còn được gọi là *đỉnh treo*.



Bậc của các đỉnh trong đồ thị trên được điền trong bảng dưới đây:

Đỉnh	A	B	C	D	E	F
Bậc	2	4	1	3	2	0

trong đó C là đỉnh treo còn F là đỉnh cô lập.

Với từng đỉnh, bậc của chúng có thể cao thấp khác nhau, có thể chẵn, có thể lẻ. Tuy nhiên khi lấy tổng tất cả lại, ta có tính chất đặc biệt dưới đây:

- **Định lý.** Trong một đồ thị vô hướng, tổng bậc của tất cả các đỉnh bằng hai lần số cạnh của nó.

  Để chứng minh chỉ cần nhận xét rằng, khi cộng tất cả các bậc của các đỉnh, mỗi cạnh của đồ thị được tính hai lần.

  Trong ví dụ trên, tổng bậc của tất cả các đỉnh bằng 12, vậy đồ thị có 6 cạnh.

  Từ định lý trên, trực tiếp suy ra hệ quả:

  - **Hệ quả 1.** Trong một đồ thị vô hướng, tổng bậc của tất cả các đỉnh bao giờ cũng là số chẵn.

    Hệ quả này có thể phát biểu cách khác, dưới dạng kết quả của bài toán tồn tại: "không tồn tại một đồ thị vô hướng nào mà tổng bậc của tất cả các đỉnh của nó là số lẻ". Kết quả đơn giản này có thể làm cơ sở cho nhiều phép phản chứng.

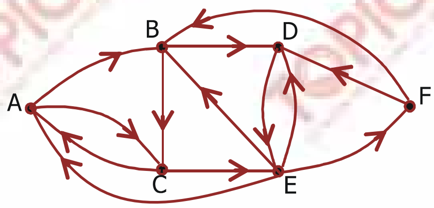
    Chẳng hạn, ta có thể khẳng định rằng, không thể nối 31 máy tính thành một mạng, trong đó mỗi máy được nối với đúng 3 máy khác. Thật vậy, nếu điều đó xảy ra, ta nhận được một đồ thị vô hướng gồm 31 đỉnh với tổng bậc là $31 \times 3 = 93$, và đấy là điều vô lý.

    Một hệ quả trực tiếp khác, suy từ hệ quả trên, được dành cho bạn đọc tự lập luận

  - **Hệ quả 2.** Trong một đồ thị vô hướng, số các đỉnh bậc lẻ, bao giờ cũng là số chẵn (hoặc không tồn tại một đồ thị vô hướng nào mà số đỉnh bậc lẻ của nó là một số lẻ).

- **Bậc của đỉnh trong đồ thị có hướng.** Trong đồ thị có hướng, khi tính bậc, người ta muốn phân biệt các cạnh vào, ra tại một đỉnh, vì thế khái niệm bậc được thay bằng khái niệm nửa bậc như định nghĩa dưới đây.

  Ta gọi *nửa bậc vào* của đỉnh u, ký hiệu $deg^{+}(u)$, là số cạnh từ ngoài đi vào u, và *nửa bậc ra* của u, ký hiệu $deg^{-}(u)$, là số cạnh từ u đi ra.



Ví dụ trong đồ thị trên, các nửa bậc vào và ra của các đỉnh, được điền trong bảng dưới đây:

Đỉnh	A	B	C	D	E	F
Nửa bậc vào	2	3	2	3	2	1
Nửa bậc ra	2	2	2	1	4	2

Có những đỉnh, nửa bậc vào bằng nửa bậc ra (A, C), có những đỉnh, nửa bậc vào lớn hơn nửa bậc ra (B, D), có những đỉnh, nửa bậc vào nhỏ hơn nửa bậc ra (E, F), nhưng cộng tất cả các nửa bậc này lại, ta có định lý:

Định lý. Trong một đồ thị có hướng, tổng nửa bậc vào của tất cả các đỉnh bằng tổng nửa bậc ra của tất cả các đỉnh và bằng số cạnh của nó.

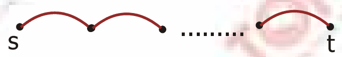
Bạn đọc dễ dàng chứng minh định lý này. Trong ví dụ trên, con số này bằng 13 là số cạnh của đồ thị.

5.2.4. Đường đi.

Ta gọi một *đường đi* từ đỉnh s đến đỉnh t là một dãy các cạnh kề nhau (cạnh sau kề cạnh trước), xuất phát từ s và kết thúc tại t.

Trong định nghĩa trên, đường đi là có hướng. Đỉnh s được gọi là *đỉnh đầu* và đỉnh t được gọi là *đỉnh cuối* của đường đi.

Nếu không xét hướng trên các cạnh, đường đi từ s đến t cũng là đường đi từ t đến s. Một đường đi vô hướng như thế, được gọi ngắn gọn là đường đi giữa hai đỉnh s và t.

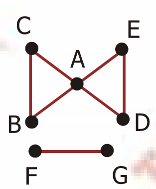


Số cạnh trên đường đi, được gọi là *độ dài* của đường đi (trong bài 8, ta sẽ gặp một định nghĩa tổng quát hơn về độ dài). Ta xem một đường đi có độ dài 0 là đường đi từ một đỉnh đến chính nó không đi qua cạnh nào, đường đi này gọi là *đường đi rỗng*.

Người ta thường quan tâm dãy các đỉnh (theo thứ tự) mà đường đi đi qua. Trong đơn đồ thị, một đường đi được xác định duy nhất từ dãy các đỉnh này, vì thế người ta thường viết một đường đi dưới dạng một dãy các đỉnh như vậy.

Giống như các tình huống xảy ra đối với các cạnh, giữa hai đỉnh có thể không có, có một, hoặc có nhiều đường đi nối chúng. Trên đường đi, một đỉnh hoặc một cạnh có thể gặp nhiều lần (chúng được gọi tương ứng là các *đỉnh lặp*, *cạnh lặp*). Một đường đi không có đỉnh lặp (điều này kéo theo không có cạnh lặp) được gọi là một *đường đi đơn*. Có thể dễ dàng chứng minh rằng, nếu có một đường đi từ s đến t thì cũng sẽ có một đường đi đơn từ s đến t. Vì thế đường đi đơn được xem như đại diện cho các đường đi khác, một số vấn đề được đặt ra chỉ có ý nghĩa đối với các đường đi đơn chẳng hạn các bài toán tìm đường đi, liệt kê đường đi, đếm đường đi, v.v...

Ví dụ: Trong đồ thị 7 đỉnh A, B, C, D, E, F, G (hình bên cạnh), từ A không có một đường đi nào đến các đỉnh F và G. Các đường đi từ A đến B có thể là AB, ACB, ADEAB, AEDACB, ... Trong các đường đi này, các đường AB, ACB là các đường đi đơn, các đường khác không phải là các đường đi đơn.



5.2.5. Chu trình

Một đường đi, không có cạnh lặp, có đỉnh đầu trùng với đỉnh cuối được gọi là một *chu trình*. Như thế, một chu trình là một đường đi khép kín, vì vậy trên chu trình, điểm đầu, điểm cuối là không quan trọng. Giống như đường đi, chu trình có thể có hướng hoặc vô hướng. Một chu trình không có đỉnh lặp (trừ đỉnh đầu, đỉnh cuối trùng nhau) được gọi là *một chu trình đơn*.

Trong Ví dụ mục trước, đồ thị có các chu trình ABCA, ADEA là các chu trình đơn, chu trình ABCADEA không phải là chu trình đơn.

Điều kiện không có cạnh lặp trong chu trình cho thấy rằng, một chu trình của một đơn đồ thị vô hướng ít nhất phải có 3 cạnh. Điều này là không đúng đối với đa đồ thị hoặc đồ thị có hướng (trong những đồ thị này, có thể có chu trình ít hơn 3 cạnh).

Đường đi và chu trình có thể xem là những cấu hình tổ hợp được tạo nên từ các cạnh của đồ thị. Sự tồn tại của chúng đã dẫn đến nhiều khái niệm quan trọng của lý thuyết đồ thị. Các bài toán tổ hợp như đếm, liệt kê, tồn tại, tối ưu đối với những cấu hình này có nhiều ý nghĩa trong các ứng dụng.

5.2.6. Liên thông

Khái niệm liên thông được xây dựng trên cơ sở tồn tại các đường đi, một đường đi vô hướng có thể trở thành không phải là đường đi khi đưa hướng vào, vì thế có phân biệt đôi chút giữa đồ thị vô hướng và đồ thị có hướng.

- **Liên thông trên đồ thị vô hướng.** Một đồ thị vô hướng được gọi là *liên thông* nếu giữa hai đỉnh bất kỳ của nó đều có đường đi (nghĩa là từ một đỉnh bất kỳ, ta có thể đến một đỉnh bất kỳ khác bằng một đường đi nào đó).

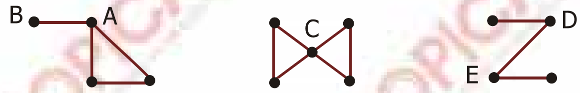
  Tính chất liên thông của đồ thị là điều kiện quan trọng trong nhiều ứng dụng. Một mạng giao thông không liên thông hoặc một mạng máy tính không liên thông có nghĩa là cần sửa chữa.

  Ta xem một đỉnh luôn có đường đi đến chính nó là đường đi rỗng (không có cạnh nào). Nói riêng đồ thị chỉ có một đỉnh luôn được xem là đồ thị liên thông.

  Trên tập đỉnh V của một đồ thị vô hướng bất kỳ G, ta xác định một quan hệ hai ngôi R bằng định nghĩa uRv (đỉnh u có quan hệ R với đỉnh v) khi và chỉ khi có đường đi nối hai đỉnh này. Dễ dàng thử lại R là quan hệ tương đương:

  - R có tính phản xạ: đường đi rỗng là đường đi nối một đỉnh với chính nó.
  - R có tính đối xứng: vì vô hướng nên vai trò của u và v trên đường đi nối chúng là như nhau.
  - R có tính bắc cầu: nếu có đường đi nối u với v và có đường đi nối v với w thì đường đi ghép hai đường này tại đỉnh v là đường đi nối u với w.

  Như vậy R sẽ phân hoạch V thành từng lớp (xem tính chất của quan hệ tương đương trong bài 1), trong đó hai đỉnh thuộc cùng một lớp là đi được đến nhau còn hai đỉnh thuộc hai lớp khác nhau là không đi được đến nhau. Mỗi lớp như vậy xác định một đồ thị con liên thông của G và được gọi là một thành phần liên thông của G.



Đồ thị G gồm 3 thành phần liên thông

Từ đó ta cũng nhận được một đồ thị vô hướng liên thông khi và chỉ khi số thành phần liên thông của nó bằng 1.

Do các thành phần liên thông của G là rời nhau và hợp của nó bằng G nên nhiều bài toán trên G được đưa về việc giải trên từng thành phần liên thông rồi ghép lại (nguyên lý cộng).

Một đỉnh mà khi xóa nó cùng với những cạnh liên thuộc ra khỏi đồ thị, ta nhận được số thành phần liên thông của đồ thị tăng lên, được gọi là *đỉnh khớp*. Tương tự một cạnh mà khi xóa nó ra khỏi đồ thị (chú ý không xóa đỉnh) làm cho số thành phần liên thông của đồ thị tăng lên, được gọi là *cầu*. Việc xóa đỉnh khớp hay xóa cầu của một đồ thị liên thông sẽ làm cho đồ thị trở nên không liên thông nữa. Đỉnh khớp và cầu mô tả những nút và đường quan trọng trong một mạng giao thông mà việc ách tắc cục bộ ở những nơi này sẽ gây ra ách tắc trên diện rộng.

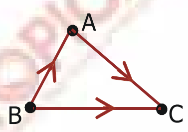
Có thể chứng minh dễ dàng rằng, một cạnh là một cầu khi và chỉ khi nó không nằm trên một chu trình nào cả.

Với đồ thị G cho ở hình vẽ trên, ta có thể kể ra tất cả các đỉnh khớp và cầu của nó bằng cách xét trên từng thành phần liên thông. Trong thành phần thứ nhất (chứa đỉnh A) ta có một đỉnh khớp là A và một cầu là AB, trong thành phần thứ hai (chứa đỉnh C) ta có một đỉnh khớp là C và không có cầu (tất cả các cạnh đều có chu trình đi qua), trong thành phần thứ ba (chứa đỉnh D) ta có hai đỉnh khớp là D, E và tất cả ba cạnh của nó đều là cầu (không có chu trình nào).

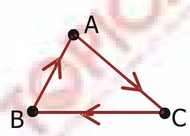
- **Liên thông trên đồ thị có hướng.** Với đồ thị có hướng, định nghĩa liên thông được chia thành hai mức: liên thông yếu và liên thông mạnh.

  Một đồ thị có hướng được gọi là *liên thông yếu* nếu nó liên thông theo nghĩa vô hướng (nghĩa là bỏ hướng đi ta được một đồ thị vô hướng liên thông), và được gọi là *liên thông mạnh* nếu nó liên thông theo nghĩa có hướng (nghĩa là phải đi theo các đường đi có hướng). Rõ ràng một đồ thị liên thông mạnh phải thỏa mãn điều kiện liên thông yếu.

  Như vậy, một đồ thị có hướng có phải là liên thông mạnh hay không, ngoài việc phụ thuộc vào đồ thị vô hướng tương ứng với nó phải liên thông, còn phụ thuộc vào hướng trên các cạnh. Ví dụ dưới đây mô tả tam giác ABC, với cách định hướng trên các cạnh là khác nhau, ta được đồ thị G_1 là liên thông yếu còn đồ thị G_2 là liên thông mạnh.



Đồ thị G_1 liên thông yếu



Đồ thị G_2 liên thông mạnh

5.2.7. Đẳng cấu

Cùng một đồ thị nhưng khi biểu diễn trên mặt phẳng, vị trí các đỉnh của nó có thể đặt tùy ý, ngoài ra có rất nhiều cách đặt tên khác nhau cho các đỉnh, điều này dẫn đến một đồ thị có những biểu diễn khác nhau. Cần phải có một định nghĩa để dựa vào đó ta biết được khi nào thì hai biểu diễn khác nhau thực sự là của cùng một đồ thị.

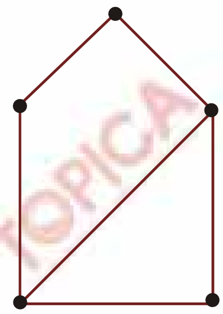
Định nghĩa. Hai đơn đồ thị G = (V, E) và G' = (V', E') được gọi là *đẳng cấu* với nhau nếu tồn tại tương ứng 1−1 giữa V và V' sao cho tương ứng này bảo toàn các cạnh, nghĩa là cặp đỉnh (u, v), u, v thuộc V, là một cạnh của G khi và chỉ khi cặp đỉnh tương ứng (u', v'), u', v' thuộc V', là một cạnh của G'.

Từ định nghĩa suy ra mọi tính chất của G đều được bảo toàn khi qua một phép đẳng cấu, chúng được gọi là các *bất biến*. Dưới đây là một số bất biến đơn giản, được dùng làm điều kiện cần cho hai đồ thị là đẳng cấu:

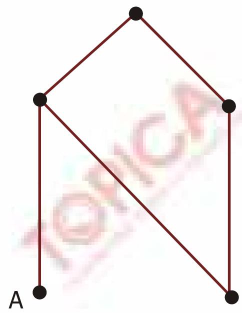
- Số đỉnh và số cạnh của đồ thị này phải bằng số đỉnh và số cạnh của đồ thị kia
- Số đỉnh bậc k của đồ thị này phải bằng số đỉnh bậc k của đồ thị kia
- Số cầu của đồ thị này phải bằng số cầu của đồ thị kia
- Số đường đi đơn (chu trình đơn) độ dài k của đồ thị này phải bằng số đường đi đơn (chu trình đơn) độ dài k của đồ thị kia…

Nếu kiểm tra có một bất biến nào đó không thỏa mãn thì có thể khẳng định hai đồ thị đang xét là không đẳng cấu. Ngược lại, nếu các bất biến được thử đều thỏa mãn thì chưa thể suy ra được hai đồ thị là đẳng cấu. Nói chung việc tìm một tương ứng 1−1 thỏa mãn điều kiện đẳng cấu giữa hai đồ thị n đỉnh là rất khó vì phải tìm chúng trong n! song ánh có thể có. Tuy nhiên, những bất biến giúp ta thu hẹp diện lựa chọn, trong một số tình huống, với n không lớn lắm, có thể trả lời được câu hỏi hai đồ thị đã cho là có đẳng cấu hay không.

Ví dụ 1. Xét hai đồ thị dưới đây xem chúng có đẳng cấu không?



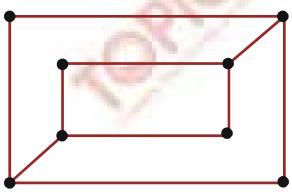
Đồ thị G



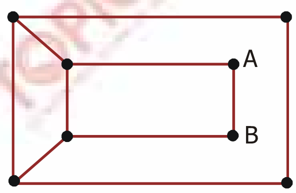
Đồ thị G'

Giải. Hai đồ thị đã cho là không đẳng cấu vì trong G' có đỉnh A bậc 1 còn trong G không có đỉnh nào bậc 1.

Ví dụ 2. Xét hai đồ thị dưới đây xem chúng có đẳng cấu không?



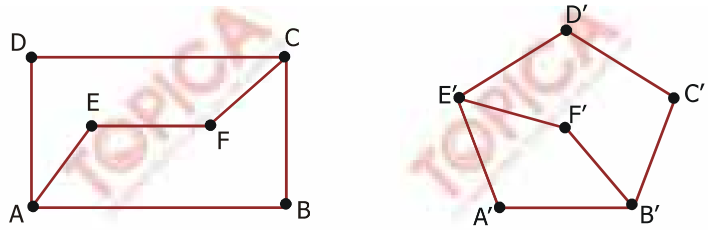
Đồ thị G



Đồ thị G'

Giải. Hai đồ thị G, G' đều có 8 đỉnh, 10 cạnh, số các đỉnh cùng bậc là như nhau (4 đỉnh bậc 2 và 4 đỉnh bậc 3) tuy nhiên chúng không đẳng cấu vì trong G' có cạnh AB nối hai đỉnh bậc 2, trong khi trong G không có một cạnh nào như vậy.

Ví dụ 3. Xét hai đồ thị dưới đây xem chúng có đẳng cấu không?



Giải. Hai đồ thị G, G' đều có 6 đỉnh, 7 cạnh, số các đỉnh cùng bậc là như nhau (gồm 4 đỉnh bậc 2 tương ứng là B, D, E, F và A', F', C', D', 2 đỉnh bậc 3 tương ứng là A, C và B', E'). Các đỉnh bậc 3 của hai đồ thị đều nối với các đỉnh bậc 2. Các chu trình đơn đều được bảo toàn (2 chu trình đơn độ dài 5 và 1 chu trình đơn độ dài 4).

Kết hợp các bất biến này, ta xác định dần các tương ứng. Trước hết 2 đỉnh bậc 3 phải tương ứng với nhau và 2 đỉnh bậc 2 trên chu trình đơn độ dài 4 phải tương ứng với nhau, ta được: A↔B', C↔E', B↔A', D↔F'. Sau đó là 2 đỉnh bậc 2 còn lại tương ứng với nhau: E↔C', F↔D'. Cuối cùng là việc thử lại các tương ứng này bảo toàn các cạnh bằng cách xét tất cả các cặp đỉnh của G và các tương ứng của chúng trong G'.

Trong Ví dụ này số các cặp đỉnh được thử là 15 cặp và trạng thái của những cặp đỉnh tương ứng này (là cạnh/không phải là cạnh) là như nhau trên hai đồ thị. Vậy hai đồ thị đã cho là đẳng cấu.

Chú ý: Có thể có nhiều tương ứng đẳng cấu khác nhau. Nếu một tương ứng được thử không thỏa mãn đẳng cấu, thì chưa thể kết luận được hai đồ thị đã cho là không đẳng cấu, kết luận này chỉ được khẳng định khi thử tất cả các tương ứng đều không thỏa mãn.

5.3. Biểu diễn đồ thị trên máy tính

Trong mục này, ta sẽ thảo luận các cách số hóa đồ thị để có thể lưu trữ trên máy tính. Ta chỉ hạn chế trên các đơn đồ thị, mặc dù nhiều vấn đề có thể mở rộng cho một đa đồ thị bất kỳ và cũng chỉ xét một số dạng thường dùng nhất.

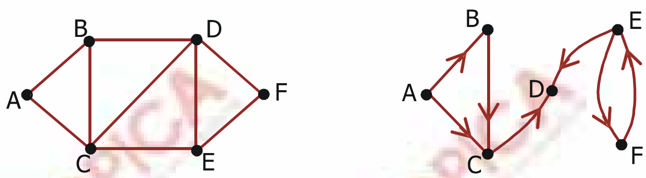
5.3.1. Ma trận kề

Giả sử các đỉnh của đồ thị G được đánh số thứ tự từ 1 đến n. Lập ma trận vuông A n dòng, n cột theo quy tắc sau: phần tử dòng i, cột j của ma trận A bằng 1 nếu đỉnh j kề với đỉnh i (nghĩa là có cạnh nối i với j) và bằng 0 nếu trái lại.

Ma trận A được gọi là *ma trận kề* của đồ thị G. Các phần tử của A chỉ nhận hai giá trị 0, 1 (người ta gọi các ma trận như thế là ma trận 0/1 hay ma trận Boole).

Định nghĩa này bao gồm cả có hướng và vô hướng tùy theo quan hệ kề có được kể hướng hay không.

Ví dụ: Xét đơn đồ thị G_1 vô hướng và đơn đồ thị G_2 có hướng dưới đây (các đỉnh của nó được gán tên A, B, C, ... thay vì đánh số):



Đồ thị vô hướng G_1 **Đồ thị có hướng G_2**

Các ma trận kề của chúng được cho tương ứng bởi các bảng dưới đây:

	A	B	C	D	E	F
A	0	1	1	0	0	0
B	1	0	1	1	0	0
C	1	1	0	1	1	0
D	0	1	1	0	1	1
E	0	0	1	1	0	1
F	0	0	0	1	1	0

Ma trận kề của G_1

	A	B	C	D	E	F
A	0	1	1	0	0	0
B	0	0	1	0	0	0
C	0	0	0	1	0	0
D	0	0	0	0	0	0
E	0	0	0	1	0	1
F	0	0	0	0	1	0

Ma trận kề của G_2

Như thế mỗi đơn đồ thị xác định một ma trận 0/1 (sai khác nhau bởi đánh số thứ tự đỉnh) là ma trận kề của nó và ngược lại, mỗi ma trận 0/1 xác định một đơn đồ thị nhận nó làm ma trận kề.

Dưới đây là một số chú ý:

- Các phần tử trên đường chéo chính của ma trận kề đều bằng không vì đồ thị không có khuyên.
- Ma trận kề của đồ thị vô hướng là ma trận đối xứng, vì cạnh của nó không phụ thuộc thứ tự đỉnh, vì thế với đồ thị vô hướng người ta chỉ cần lưu trữ một nửa của ma trận kề (chẳng hạn phía trên đường chéo chính), điều này là không đúng cho trường hợp đồ thị có hướng.

Ma trận kề là cách lưu trữ đơn giản và thuận tiện cho nhiều xử lý trên đồ thị, tốc độ tìm kiếm nhanh, trong máy tính chúng được cài đặt bằng mảng hai chiều, một kiểu dữ liệu quen thuộc trong nhiều ngôn ngữ. Nhiều thông tin của đồ thị được tính trực tiếp dễ dàng từ ma trận kề, chẳng hạn tính số cạnh của đồ thị hay tính bậc của mỗi đỉnh, ngoài ra các phép toán trên ma trận là những công cụ hữu ích giúp cho việc trình bày các kết quả cũng như các thuật toán trên đồ thị.

Nhờ ma trận kề, ta có thể kiểm chứng, một cách minh bạch và không nhầm lẫn, một tương ứng giữa hai đồ thị có phải là đẳng cấu hay không. Để làm điều này ta lập hai ma trận kề của hai đồ thị đang xét theo thứ tự tương ứng của các đỉnh, sau đó so sánh hai ma trận này. Nếu chúng trùng nhau thì tương ứng đang xét là đẳng cấu, nếu ngược lại cần tìm tương ứng khác.

Quay lại Ví dụ 3 trong mục 5.2.6, với thứ tự tương ứng A↔B’, B↔A’, C↔E’, D↔F’, E↔C’, F↔D’, ta được các ma trận kề của G và G’ (chỉ viết một nửa vì các ma trận là đối xứng):

	A	B	C	D	E	F
A	0	1	0	1	1	0
B		0	1	0	0	0
C			0	1	0	1
D				0	0	0
E					0	1
F						0

Ma trận kề của G

	B′	A′	E′	F′	C′	D′
B′	0	1	0	1	1	0
A′		0	1	0	0	0
E′			0	1	0	1
F′				0	0	0
C′					0	1
D'						0

Ma trận kề của G′

Hai ma trận này bằng nhau, vậy tương ứng đang xét là đẳng cấu.

Hạn chế rõ nét nhất của cách lưu trữ ma trận kề là lãng phí bộ nhớ, đặc biệt với những đồ thị ít cạnh. Ngoài ra cách lưu trữ này không linh hoạt và thuận tiện trong những xử lý mà đồ thị có sự thay đổi phát sinh (thêm, bớt các đỉnh, cạnh). Thông thường cách biểu diễn đồ thị bằng ma trận kề được áp dụng cho các đồ thị ít đỉnh và việc cài đặt được dùng cho việc tìm hiểu thuật toán.

5.3.2. Danh sách cạnh

Tập hợp tất cả các cạnh của đơn đồ thị G, mỗi cạnh được biểu diễn như một cặp đỉnh, xếp theo một thứ tự nào đấy, được gọi là *danh sách cạnh* của G. Tùy thuộc vào các cặp đỉnh có kể thứ tự hay không mà đồ thị G được hiểu là có hướng hay vô hướng.

Như vậy, một đơn đồ thị xác định một danh sách cạnh (sai khác nhau thứ tự các cạnh trong danh sách) và ngược lại mỗi danh sách cạnh xác định một đơn đồ thị.

Với hai đơn đồ thị G_1 và G_2 trong Ví dụ mục 5.3.1, các danh sách cạnh của chúng là:

Đồ thị G_1: {AB, AC, BC, BD, CD, CE, DE, DF, EF} (9 cạnh vô hướng).

Đồ thị G_2: {AB, AC, BC, CD, ED, EF, FE} (7 cạnh có hướng).

Việc lưu trữ danh sách cạnh đã chú ý tiết kiệm bộ nhớ hơn so với lưu trữ ma trận kề, nhưng việc tính toán thường chậm và phức tạp hơn. Chẳng hạn mỗi lần muốn kiểm tra đỉnh v có kề với đỉnh u không là một lần phải duyệt danh sách cạnh.

Để cài đặt danh sách cạnh, người ta thường dùng một mảng các bản ghi (trong Pascal) hoặc mảng các cấu trúc (trong C), trong đó mỗi phần tử mảng lưu trữ một cạnh, gồm hai trường ghi tên hai đỉnh tương ứng.

Cách biểu diễn đồ thị bằng danh sách cạnh thường được dùng trong các bài toán mà việc xử lý chủ yếu trên các đối tượng cạnh.

5.3.3. Danh sách kề

Với mỗi đỉnh u của đơn đồ thị G, ta xác định các đỉnh kề với nó. Danh sách này được gọi là *danh sách kề* của đỉnh u. Tập hợp các danh sách này với mọi đỉnh u của G, được gọi là *danh sách kề* của G. Như thế mỗi đơn đồ thị xác định một danh sách kề (sai khác nhau thứ tự các phần tử trong danh sách). Ngược lại, mỗi danh sách kề xác định một đơn đồ thị.

Với hai đơn đồ thị G_1 và G_2 trong Ví dụ mục 5.3.1, các danh sách kề của chúng là:

Danh sách kề của đồ thị G_1 bao gồm:

- Danh sách kề đỉnh A: {B, C}
- Danh sách kề đỉnh B: {A, C, D}
- Danh sách kề đỉnh C: {A, B, D, E}
- Danh sách kề đỉnh D: {B, C, E, F}
- Danh sách kề đỉnh E: {C, D, F}
- Danh sách kề đỉnh F: {D, E}

Danh sách kề của đồ thị G_2 bao gồm:

- Danh sách kề đỉnh A: {B, C}
- Danh sách kề đỉnh B: {C}

- Danh sách kề đỉnh C: {D}
- Danh sách kề đỉnh D: {} (rỗng)
- Danh sách kề đỉnh E: {D, F}
- Danh sách kề đỉnh F: {E}

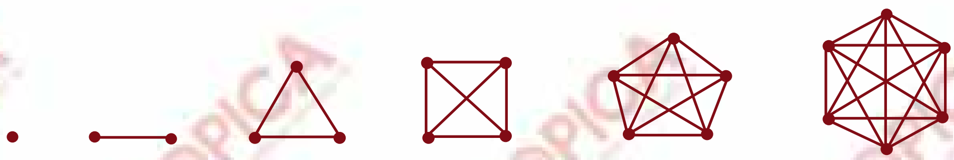
Các danh sách này thường được cài đặt bằng cấp phát động dưới dạng một danh sách liên kết. Với cách tổ chức này, biểu diễn đồ thị bằng danh sách kề là phương pháp tận dụng tốt nhất bộ nhớ, đặc biệt với các đồ thị nhiều đỉnh nhưng ít cạnh. Ngoài ra đây là cách thích hợp cho các bài toán mà trong đó luôn gặp các thao tác biến đổi đồ thị như thêm, bớt các đỉnh, cạnh. Tuy nhiên, cách này thường phức tạp vì phải dùng các con trỏ để quản lý dữ liệu, dễ gây ra nhầm lẫn trong việc dùng các ngôn ngữ lập trình.

5.4. Một số đồ thị đặc biệt

Trong mục này ta sẽ xét một số đồ thị đặc biệt với các tên gọi riêng, chúng sẽ có ích trong việc minh họa một số khái niệm, kết quả và ứng dụng của lý thuyết đồ thị. Ta chỉ giới hạn các định nghĩa trong lớp các đồ thị vô hướng, mặc dù trong một số tình huống, các định nghĩa này có thể mở rộng cho lớp các đồ thị có hướng.

5.4.1. Đồ thị đầy đủ

Một đơn đồ thị vô hướng mà tất cả các cặp đỉnh đều được nối với nhau, được gọi là một đồ thị đầy đủ. Như thế một đồ thị đầy đủ hoàn toàn được xác định khi biết số đỉnh của nó. Đồ thị đầy đủ n đỉnh được ký hiệu là K_n.

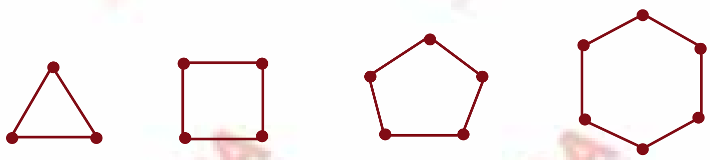


K_1 K_2 K_3 K_4 K_5 K_6

Số cạnh của K_n là $n(n-1)/2$, mọi đỉnh đều có bậc bằng nhau và bằng $n-1$. Theo định nghĩa, K_n là đồ thị có nhiều cạnh nhất trong lớp các đơn đồ thị vô hướng n đỉnh.

5.4.2. Đồ thị vòng

Một đơn đồ thị vô hướng có một chu trình đơn đi qua tất cả các đỉnh và các cạnh của nó được gọi là *đồ thị vòng*. Như thế một đồ thị vòng sẽ trùng với chính chu trình này và nó được hoàn toàn xác định khi biết số đỉnh (không dưới 3). Đồ thị vòng n đỉnh được ký hiệu là C_n.

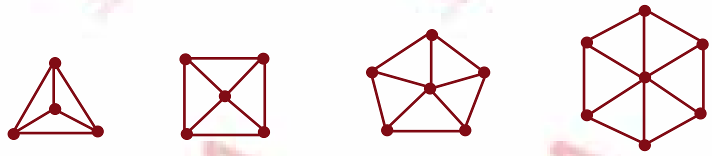


C_3 C_4 C_5 C_6

Số cạnh của C_n bằng n, mọi đỉnh đều có bậc bằng nhau và bằng 2. Đồ thị C_3 trùng với đồ thị K_3.

5.4.3. Đồ thị bánh xe

Thêm vào đồ thị vòng C_n một đỉnh mới và nối nó với tất cả các đỉnh khác, ta thu được một đồ thị n+1 đỉnh, gọi là *đồ thị bánh xe*, ký hiệu là W_n.

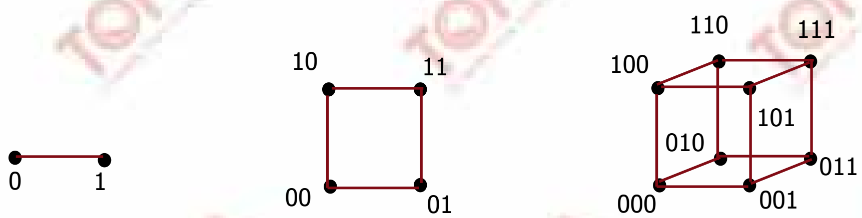


W_3 **W_4** **W_5** **W_6**

Số cạnh của W_n bằng 2n. Đỉnh mới thêm (được gọi là *đỉnh tâm*) có bậc là n, các đỉnh khác (gọi là các *đỉnh biên*) đều có bậc bằng nhau và bằng 3. Đồ thị W_3 trùng với đồ thị K_4.

5.4.4. Đồ thị lập phương

Xây dựng một đơn đồ thị vô hướng gồm 2^n đỉnh, mỗi đỉnh được đặt tên bằng một chuỗi n bit. Hai đỉnh được nối với nhau nếu và chỉ nếu tên của chúng chỉ khác nhau đúng một bit. Đồ thị được xây dựng, được gọi là *đồ thị lập phương*, n được gọi là *số chiều*, ký hiệu là Q_n.



Q_1 **Q_2** **Q_3**

Các đỉnh của Q_n đều có bậc bằng nhau và bằng n, từ đó nhận được số cạnh của Q_n là $n.2^n/2 = n.2^{n-1}$.

Các đồ thị K_n, C_n, Q_n đều có một tính chất là bậc của các đỉnh đều bằng nhau. Người ta gọi chung các đơn đồ thị vô hướng có tính chất như vậy là các *đồ thị chính quy*, giá trị bậc của các đỉnh của đồ thị chính quy được gọi là bậc của đồ thị đó. Chẳng hạn K_n có bậc n−1, C_n có bậc 2, Q_n có bậc n.

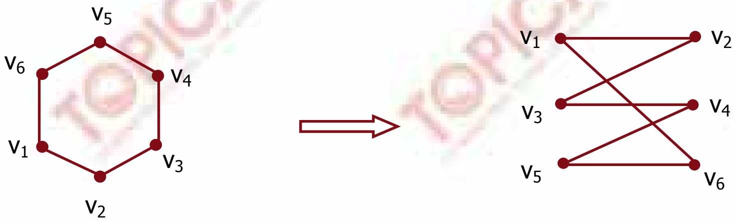
Các đồ thị K_n, C_n, W_n, Q_n đều là những đồ thị liên thông, chúng thường được dùng trong việc tổ chức kết nối các bộ xử lý trong một mạng máy tính.

5.4.5. Đồ thị hai phía

Định nghĩa: Một đơn đồ thị vô hướng được gọi là *hai phía* nếu có thể phân hoạch tập đỉnh thành hai lớp sao cho chỉ có các cạnh nối đỉnh của lớp này với đỉnh của lớp kia (nghĩa là không có cạnh nối hai đỉnh của cùng một lớp).

Ví dụ: đồ thị vòng C_6 là một đồ thị hai phía. Thật vậy, đánh số các đỉnh theo thứ tự trên chu trình $v_1, v_2, ..., v_6$. Khi đó mỗi cạnh của C_6 sẽ nối một đỉnh có chỉ số lẻ với một đỉnh có chỉ số chẵn, từ đó nhận được phân hoạch tập đỉnh thành hai lớp $\{v_1, v_3, v_5\}$, $\{v_2, v_4, v_6\}$ thỏa mãn điều kiện hai phía.

Hình vẽ dưới đây trình bày lại C_6 dưới dạng hai phía:

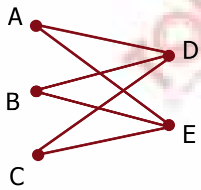


Để khẳng định một đơn đồ thị vô hướng không phải là hai phía ta phải chứng minh mọi phân hoạch tập đỉnh thành hai lớp, đều có ít nhất một cạnh nối hai đỉnh thuộc cùng một lớp. Chẳng hạn K_3 không phải là đồ thị hai phía vì mọi phân hoạch tập đỉnh thành hai lớp đều có một lớp chứa hai đỉnh do đó xuất hiện một cạnh nối hai đỉnh này (lập luận này vẫn đúng với mọi đồ thị đầy đủ K_n, $n \geq 3$).

Việc chứng minh định lý dưới đây xem như bài tập:

Định lý. Đơn đồ thị vô hướng là đồ thị hai phía khi và chỉ khi nó không chứa chu trình độ dài lẻ.

Một đồ thị hai phía có tập đỉnh được phân hoạch thành một lớp *m* đỉnh, một lớp *n* đỉnh sao cho mọi cặp đỉnh khả dĩ (một đỉnh thuộc lớp này, một đỉnh thuộc lớp kia) đều có cạnh nối, được gọi là đồ thị *hai phía đầy đủ*, ký hiệu là $K_{m,\,n}$.



Đồ thị hai phía đầy đủ $K_{3,\,2}$

Số cạnh của $K_{m,\,n}$ bằng m.n, các đỉnh thuộc lớp m đỉnh có bậc n, các đỉnh thuộc lớp n đỉnh có bậc m.

Đồ thị hai phía dùng để mô tả mối quan hệ giữa các phần tử thuộc hai tập khác nhau, chẳng hạn các trận đấu cá nhân giữa hai đội cờ, các bắt tay giữa hai phái đoàn, mối quan hệ phù hợp giữa một nhóm người và một nhóm nghề nghiệp, ... Đồ thị hai phía đầy đủ $K_{3,2}$ mô tả các trận đấu giữa hai đội cờ, một đội 3 ngguời, một đội 2 người, sao cho mỗi người của đội này đều gặp tất cả mọi người của đội kia, mỗi người đúng một trận.

Nhiều ứng dụng, trong đó cần phải ghép một phần tử của tập hợp này với một phần tử của tập hợp khác thỏa mãn một quan hệ nào đấy (bài toán về các cặp ghép), được đưa về đồ thị hai phía và người ta đã xây dựng được những thuật toán hữu hiệu trên đồ thị này để giải quyết.

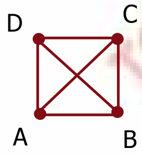
5.4.6. Đồ thị phẳng

Khi biểu diễn đồ thị trên mặt phẳng ta gặp một tình huống có những cạnh cắt nhau tại những giao điểm không phải đỉnh. Một số trường hợp ta tìm được cách vẽ khác để không xảy ra tình huống vừa nêu, trong khi có những trường hợp không tồn tại một cách vẽ nào như thế. Để phân biệt các trường hợp này, người ta đưa vào định nghĩa sau:

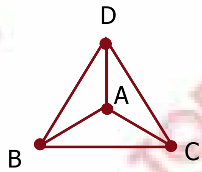
Định nghĩa. Một đồ thị được gọi là *phẳng* nếu có thể vẽ nó trên mặt phẳng sao cho không có cạnh nào cắt nhau trừ tại đỉnh. Hình vẽ như thế được gọi là một *biểu diễn phẳng* của đồ thị.

Ví dụ:

Cách vẽ K_4

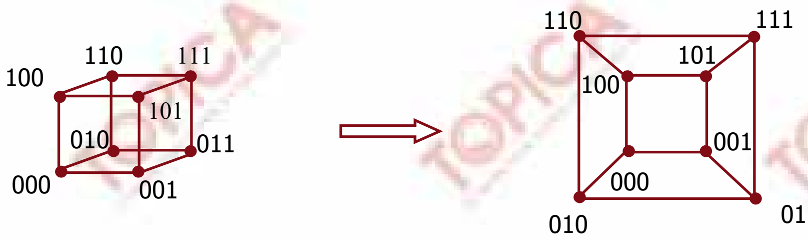


không phải là biểu diễn phẳng, tuy nhiên có thể vẽ lại K_4 một cách phẳng như sau:



Như thế K_4 là một đồ thị phẳng.

Một ví dụ nữa về đồ phẳng là đồ thị lập phương Q_3, cách vẽ như một khối lập phương có thể biểu diễn phẳng như sau:



Có thể thấy ngay rằng các đồ thị K_n ($n \le 4$), C_n, W_n là những đồ thị phẳng.

Chú ý:

Hướng của các cạnh cũng như việc có nhiều cạnh nối giữa hai đỉnh không đóng một vai trò gì trong định nghĩa đồ thị phẳng, vì thế định nghĩa phẳng chỉ cần giới hạn trong lớp đơn đồ thị vô hướng.

Tính phẳng của đồ thị được ứng dụng trong việc xây dựng các mạch điện. Nếu mạch điện là một đồ thị phẳng thì có thể thiết kế chúng sao cho không có một đường nối nào chồng lên nhau.

Để chứng tỏ một đồ thị là phẳng, ta có thể chỉ ra một biểu diễn phẳng của nó. Tuy nhiên, chứng tỏ một đồ thị không phẳng trở nên khó hơn nhiều. Người ta đã chứng minh được $K_{3,3}$ và K_5 là những đồ thị không phẳng. Chúng là những đồ thị không phẳng điển hình vì từ chúng người ta có thể nhận diện được những đồ thị không phẳng khác.

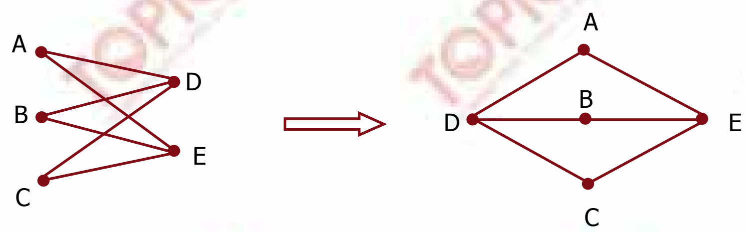
Nhà toán học người Balan, Kuratowski, đã đưa ra một định lý cho một điều kiện cần và đủ để một đồ thị là phẳng vào năm 1930, có liên quan đến các đồ thị $K_{3,3}$ và K_5. Chúng tôi không trình bày định lý này ở đây, mà chỉ nêu một số kết quả đơn giản, dễ dùng:

1. Nếu một đồ thị chứa một đồ thị con không phẳng thì nó cũng không phẳng, nói riêng các đồ thị có chứa $K_{3,3}$ hay K_5 như những đồ thị con là các đồ thị không phẳng.

2. Các đồ thị đầy đủ K_n với $n \ge 5$ là các đồ thị không phẳng.

3. Đồ thị đầy đủ hai phía $K_{m,n}$ là phẳng khi và chỉ khi $m \le 2$ hoặc $n \le 2$.

Kết quả 1 là hiển nhiên, kết quả 2 suy từ K_n, $n \ge 5$ có đồ thị con là K_5, để chứng minh kết quả 3, ta để ý rằng mọi $K_{m,n}$ với $m \ge 3$ và $n \ge 3$ đều chứa $K_{3,3}$ như đồ thị con, còn mọi $K_{1, n}$, $K_{2, n}$ (hay $K_{m, 1}$, $K_{m, 2}$) đều có thể vẽ phẳng. Chẳng hạn $K_{3, 2}$ có thể vẽ phẳng như sau:



Khi một đồ thị được biểu diễn phẳng, mặt phẳng được chia thành một số miền (trong đó có một miền vô hạn). Nhà toán học Euler đã tìm ra một công thức liên hệ giữa số đỉnh, số cạnh và số miền của một biểu diễn phẳng bất kỳ của một đơn đồ thị phẳng liên thông như sau:

Số đỉnh + Số miền − Số cạnh = 2.

Công thức trên được Euler chứng minh đầu tiên cho một hình đa diện nên nó còn có tên là *hệ thức Euler cho hình đa diện*. Mỗi hình đa diện có thể biểu diễn như một đồ thị phẳng, trong đó mỗi đỉnh của đa diện là một đỉnh của đồ thị, mỗi cạnh của đa diện là một cạnh của đồ thị, mỗi mặt của đa diện là một miền của đồ thị. Chẳng hạn hình tứ diện được biểu diễn bởi K_4, nó có 4 đỉnh, 6 cạnh và 4 mặt, hình hộp được biểu diễn bởi Q_3, nó có 8 đỉnh, 12 cạnh và 6 mặt. Hình thập nhị diện được biểu diễn bởi một đồ thị phẳng gồm 20 đỉnh, 30 cạnh và 12 mặt.

Đồ thị phẳng còn liên quan đến một bài toán nổi tiếng là bài toán bốn màu (xem mục 2.3.6 của bài 2). Một bản đồ các nước có thể xem là một đồ thị phẳng trong nó mỗi nước là một đỉnh, hai nước có chung biên giới là hai đỉnh kề nhau. Bài toán tô màu bản đồ với số mầu ít nhất được dẫn về bài toán tô màu các đỉnh của một đồ thị phẳng với số màu ít nhất sao cho không có hai đỉnh nào kề nhau có cùng màu. Năm 1890, bằng công cụ đồ thị, Heawood đã chứng minh được mọi đồ thị phẳng đều tô được bằng 5 màu, tuy nhiên mãi đến năm 1976, bài toán bốn màu mới được giải quyết bởi hai nhà toán học Mỹ là Kenneth Appel và Wolfgang Haken với sự trợ giúp của máy tính điện tử.

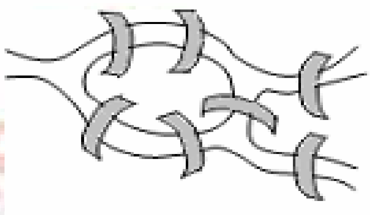
Để tìm hiểu các chứng minh chi tiết, bạn đọc có thể tham khảo [3] hoặc một số giáo trình chuyên sâu hơn về đồ thị.

Các mục tiếp theo trình bày hai loại đồ thị được mang tên hai nhà toán học nổi tiếng là Euler và Hamilton.

5.4.7. Đồ thị Euler

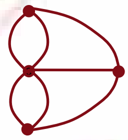
Khái niệm đồ thị Euler được hình thành từ bài toán về 7 cây cầu ở Konigberg do Leonhard Euler đưa ra vào năm 1736.

Nội dung bài toán như sau: "Thành phố Konigberg được chia thành bốn vùng bởi các nhánh sông Pregel. Người ta xây 7 cây cầu nối các vùng này với nhau để làm một công viên như hình vẽ:



Hỏi có thể xuất phát tại một nơi nào đó trong thành phố đi qua tất cả các cầu, mỗi cầu đúng một lần rồi lại trở về điểm xuất phát được không?"

Euler đã mô tả công viên như một đa đồ thị vô hướng gồm 4 đỉnh, 7 cạnh, mỗi đỉnh là một vùng đất, mỗi cạnh là một cái cầu như hình dưới đây:

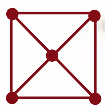


và bài toán được phát biểu lại như sau: trên đồ thị vừa vẽ, có tồn tại một chu trình đi qua tất cả các cạnh, mỗi cạnh đúng một lần hay không?

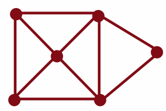
Định nghĩa. Một đồ thị vô hướng liên thông được gọi là *Euler* nếu tồn tại một chu trình đi qua tất cả các cạnh của đồ thị, mỗi cạnh đúng một lần. Chu trình tìm được, được gọi là *chu trình Euler*.

Nếu trong định nghĩa trên, ta nới rộng điều kiện chu trình bằng điều kiện đường đi (không nhất thiết khép kín) thì đồ thị được gọi là *nửa Euler* và đường đi tìm được, được gọi là *đường đi Euler*.

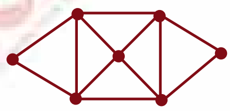
Như thế điều kiện Euler là trường hợp riêng của điều kiện nửa Euler. Nếu không thỏa mãn nửa Euler thì đồ thị được gọi là *không Euler*. Trên thực tế ta gặp cả ba tình huống: không Euler, nửa Euler và Euler như ba đồ thị dưới đây:



Không Euler



Nửa Euler



Euler

Euler đã chứng minh được định lý cho điều kiện cần và đủ của một đồ thị Euler như sau:

Định lý Euler. Một đồ thị vô hướng liên thông là Euler khi và chỉ khi tất cả các đỉnh của nó đều có bậc chẵn.

Điều kiện cần được chứng minh tương đối hiển nhiên: khi chu trình Euler đi qua một đỉnh sẽ làm tăng bậc của đỉnh đó lên 2 đơn vị, mặt khác, mỗi cạnh của đồ thị xuất hiện đúng một lần trong chu trình này, từ đó nhận được tất cả các đỉnh của đồ thị đều có bậc chẵn. Điều ngược lại được chứng minh bằng quy nạp theo số cạnh của đồ thị. Bạn đọc có thể xem chi tiết chứng minh trong [2] hoặc [3].

Từ định lý trên, ta có thể mở rộng cho đồ thị nửa Euler, xem như một hệ quả

Hệ quả. Một đồ thị vô hướng liên thông là nửa Euler khi và chỉ khi nó có không quá hai đỉnh bậc lẻ.

Chú ý: Do số đỉnh bậc lẻ của đồ thị vô hướng bao giờ cũng là số chẵn nên điều kiện có không quá hai đỉnh bậc lẻ là tương đương với hoặc không có đỉnh bậc lẻ nào hoặc có đúng hai đỉnh bậc lẻ. Trong trường hợp đầu, đồ thị là Euler, trong trường hợp sau nó chỉ là nửa Euler (không thỏa mãn Euler).

Như thế, tính chất Euler của một đồ thị vô hướng liên thông hoàn toàn được xác định nhờ việc tính bậc của các đỉnh như bảng tổng kết dưới đây:

Số đỉnh bậc lẻ	Tính chất Euler
0	Euler
2	nửa Euler
còn lại	không Euler

Dùng kết quả này để kiểm chứng ba đồ thị vừa vẽ: đồ thị đầu (hình phong bì đóng hai đầu) có bốn đỉnh bậc 3, đỉnh còn lại bậc 4, vậy nó là đồ thị không Euler, đồ thị thứ hai (hình phong bì mở một đầu) có hai đỉnh bậc 3, các đỉnh còn lại bậc chẵn, vậy nó là đồ *thị nửa Euler*, đồ thị cuối cùng (hình phong bì mở hai đầu) có các đỉnh đều bậc chẵn, vậy nó là *đồ thị Euler*.

Quay lại bài toán 7 cây cầu của Euler. Đồ thị biểu diễn nó gồm 4 đỉnh, 7 cạnh, trong đó có ba đỉnh bậc 3 và một đỉnh bậc 5. Vậy đồ thị là không Euler, và câu trả lời là không thể đi qua các cầu thỏa mãn yêu cầu của đề bài.

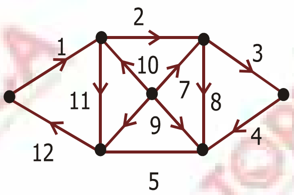
Dựa vào việc chứng minh định lý trên, người ta xây dựng được nhiều thuật toán khác nhau để tìm chu trình Euler khi điều kiện định lý Euler được thỏa mãn.

Bạn đọc có thể tham khảo [2] để tìm hiểu thuật toán, trong đó dùng stack để lưu trữ các đỉnh trong quá trình tìm chu trình, rất tiện cho việc cài đặt. Tuy nhiên thuật toán dưới đây trực quan hơn, thường được dùng khi giải thủ công.

Thuật toán Fleury. Giả sử đồ thị thỏa mãn điều kiện Euler (không có đỉnh bậc lẻ). Để tìm chu trình Euler, xuất phát từ một đỉnh bất kỳ, đi theo các cạnh của đồ thị theo hai quy tắc sau đây:

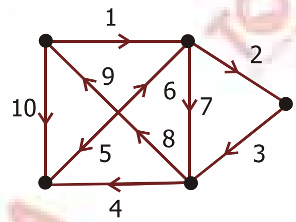
- Sau khi đi qua một cạnh, xóa cạnh đó ra khỏi đồ thị (quy tắc này đảm bảo không qua cạnh nào quá một lần).
- Chỉ đi qua cầu khi không còn con đường nào khác (quy tắc này đảm bảo đi qua tất cả các cạnh của đồ thị).

Ví dụ: trên đồ thị Euler mô tả hình phong bì mở hai đầu, một chu trình Euler tìm được nhờ thuật toán Fleury như sau:



trong đó số thứ tự đi theo chu trình được ghi trên các cạnh. Chú ý rằng, chu trình tìm được là không duy nhất vì tại mỗi bước đi ta có nhiều khả năng lựa chọn khác nhau.

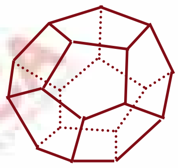
Nếu đồ thị chỉ thỏa mãn điều kiện nửa Euler (có đúng hai đỉnh bậc lẻ), thì đường đi Euler cũng được tìm theo cách như trên, trong đó chú ý đỉnh xuất phát phải là một trong hai đỉnh bậc lẻ (đỉnh bậc lẻ còn lại sẽ là đỉnh kết thúc). Ví dụ trên đồ thị nửa Euler mô tả hình phong bì mở một đầu, xuất phát từ đỉnh góc trên trái (có bậc 3), một đường đi Euler tìm được nhờ thuật toán Fleury như sau:



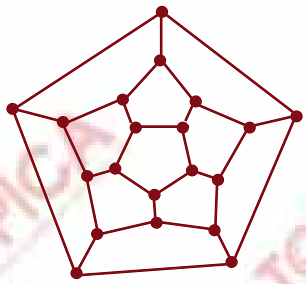
Một hình được vẽ bằng một chu trình Euler hay đường đi Euler được gọi là *vẽ được một nét* (vẽ liên tục, không nhấc bút, vẽ đủ nét và mỗi nét vẽ đúng một lần). Trước đây, trò chơi vẽ một nét, giống như trò chơi ô chữ, thường được xuất hiện trong các tuần báo như là một trò giải trí, trong đó có những hình trông rất phức tạp nhưng vẽ được một nét, đồng thời có những hình có vẻ vẽ được một nét dễ dàng nhưng không thể vẽ được. Trò chơi vẽ một nét bây giờ được hoàn toàn giải quyết bằng kết quả của Euler. Đầu tiên kiểm tra xem hình vẽ có thỏa mãn điều kiện nửa Euler hay không, nếu không thì khẳng định hình không thể vẽ được bằng một nét, trái lại hình được vẽ một nét theo thuật toán đã trình bày. Điều này giải thích tại sao từ khi kết quả của Euler trở nên phổ cập, trò chơi này không còn nữa.

5.4.8. Đồ thị Hamilton

Giống như đồ thị Euler, đồ thị với tên Hamilton xuất phát từ một bài toán vui do nhà toán học người Anh, William Rowan Hamilton đưa ra vào năm 1857. Bài toán có tên gọi "vòng quanh thế giới" với nội dung như sau:" Lấy 20 thành phố nổi tiếng trên thế giới, mỗi thành phố đặt vào một đỉnh của khối thập nhị diện đều (gồm 20 đỉnh, 12 mặt, mỗi mặt là một ngũ giác đều, xem hình vẽ dưới). Xuất phát từ một thành phố, chỉ được phép đi theo các cạnh của khối đa diện, có thể đi thăm tất cả các thành phố khác, mỗi thành phố đúng một lần, rồi lại quay về nơi xuất phát được không?"



Biểu diễn khối thập nhị diện như một đồ thị phẳng 20 đỉnh, 30 cạnh:



và bài toán "vòng quanh thế giới" của Hamilton được phát biểu lại như sau: trên đồ thị vừa vẽ có tìm được một chu trình đi qua tất cả các đỉnh, mỗi đỉnh đúng một lần hay không?

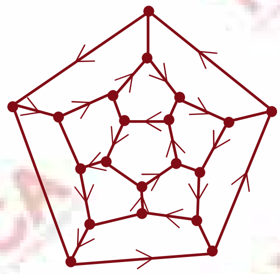
Định nghĩa. Một đồ thị vô hướng liên thông được gọi là *Hamilton* nếu tồn tại một chu trình đi qua tất cả các đỉnh của đồ thị, mỗi đỉnh đúng một lần. Chu trình tìm được, được gọi là *chu trình Hamilton*.

Nếu trong định nghĩa trên, ta nới rộng điều kiện chu trình bằng điều kiện đường đi (không nhất thiết khép kín) thì đồ thị được gọi là *nửa Hamilton* và đường đi tìm được, được gọi là *đường đi Hamilton*.

Điều kiện Hamilton xem như trường hợp riêng của nửa Hamilton. Một đồ thị không thỏa mãn điều kiện nửa Hamilton được gọi là *không Hamilton*. Cả ba tình huống Hamilton, nửa Hamilton và không Hamilton đều có thể xảy ra như ba đồ thị dưới đây:

Mặc dù về hình thức, định nghĩa về đồ thị Hamilton có thể nhận được từ định nghĩa đồ thị Euler, bằng cách thay từ "cạnh" bởi từ "đỉnh", nhưng chúng hoàn toàn khác nhau. Một chu trình đi qua tất cả các cạnh có thể qua một đỉnh nhiều lần, trong khi một chu trình đi qua tất cả các đỉnh có thể không qua một cạnh nào đó.

Đối với đồ thị Euler, vấn đề xem như được giải quyết trọn vẹn nhờ một dấu hiệu đơn giản cho bởi định lý Euler, còn đối với đồ thị Hamilton tình huống trở nên phức tạp hơn. Cho đến nay, người ta chưa tìm ra được một điều kiện cần và đủ hữu hiệu nào để nhận biết một đồ thị là Hamilton giống như định lý Euler. Trong trường hợp tổng quát nhất, thuật toán kiểm tra một đồ thị có Hamilton hay không, vẫn là thuật toán "xét mọi khả năng". Chẳng hạn bằng thuật toán quay lui, bạn đọc có thể tìm được một chu trình Hamilton trong bài toán "vòng quanh thế giới" (đi theo chiều mũi tên trên các cạnh trong hình vẽ).



Chu trình vòng quanh thế giới

Các kết quả hiện nay về đồ thị Hamilton thường được biết dưới dạng một điều kiện đủ với phần lớn mang nội dung "nếu có số cạnh đủ lớn thì đồ thị là Hamilton". Một trong những số này là định lý dưới đây.

Định lý Dirak (1952). Đơn đồ thị vô hướng liên thông với số đỉnh $n > 2$, mỗi đỉnh có bậc không nhỏ hơn $n/2$ là đồ thị Hamilton.

Ví dụ: Đồ thị đầy đủ K_n ($n > 2$), có bậc của mỗi đỉnh bằng $n-1$, thỏa mãn điều kiện Dirak, vậy nó là đồ thị Hamilton. Tuy nhiên, có nhiều đồ thị n đỉnh, có bậc của mỗi đỉnh nhỏ hơn một cách đáng kể so với $n/2$, vẫn thỏa mãn định nghĩa đồ thị Hamilton, chẳng hạn đồ thị vòng C_n với $n > 4$ (bậc của mỗi đỉnh bằng 2, nhỏ hơn $n/2$), đồ thị "vòng quanh thế giới" (bậc của mỗi đỉnh bằng 3, nhỏ hơn $20/2 = 10$), đồ thị lập phương Q_n với $n > 2$ (bậc của mỗi đỉnh bằng n, nhỏ hơn $2^n/2 = 2^{n-1}$).

Một số đồ thị, do những đặc điểm nào đấy, việc chỉ ra một chu trình (đường đi) Hamilton, hoặc phủ định sự tồn tại của nó, có những lập luận riêng, chẳng hạn để chỉ ra đồ thị lập phương Q_n là đồ thị Hamilton, người ta xây dựng một thuật toán cho phép liệt kê các chuỗi n bit theo thứ tự sao cho hai chuỗi liền kề chỉ khác nhau một bit. Tuy nhiên trong trường hợp tổng quát, việc khẳng định hoặc phủ định sự tồn tại của chu trình (đường đi) Hamilton là rất khó. Thông thường người ta phải thử mọi khả năng (chẳng hạn bằng quay lui) và phải nhờ sự hỗ trợ của máy tính.

Một số bài toán nổi tiếng như bài toán "người du lịch", bài toán "mã đi tuần", là những bài toán tìm chu trình, đường đi Hamilton, mà việc tiếp cận giải chúng đã đặt ra nhiều vấn đề cho lý thuyết đồ thị.

Các đồ thị Euler và Hamilton được mở rộng một cách tự nhiên cho đồ thị có hướng. Để tìm hiểu thêm chi tiết về các kết quả của các đồ thị này, bạn đọc có thể tham khảo [2], [3] và những tài liệu chuyên sâu về đồ thị.

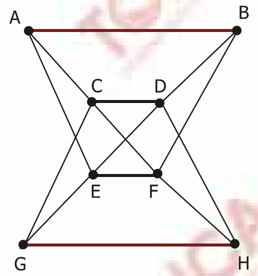
TÓM LƯỢC CUỐI BÀI

Sau khi học xong bài học này, các bạn cần ghi nhớ các vấn đề sau:

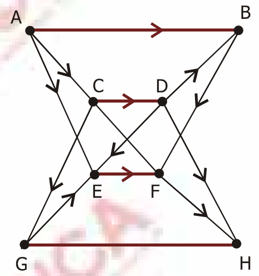
- Định nghĩa về đồ thị và các loại đồ thị.
- Các thuật ngữ như: kề, bậc của đỉnh, đường đi, chu trình, liên thông, …
- Biểu diễn đồ thị trên máy tính bằng ma trận kề, danh sách cạnh hoặc danh sách kề.
- Một số đồ thị đặc biệt và ứng dụng.

BÀI TẬP

1. Cho các đồ thị I (vô hướng), II (có hướng) bằng các hình vẽ dưới đây:



I



II

a. Xác định bậc của các đỉnh trong đồ thị I.

b. Xác đỉnh nửa bậc vào, nửa bậc ra của các đỉnh trong đồ thị II.

c. Viết các ma trận kề và danh sách kề của các đồ thị I và II.

2. Cho đồ thị có hướng với tập đỉnh là {2, 3, 4, 5, 6}. Đỉnh u được nối với đỉnh v bằng một cạnh có hướng nếu u < v và u nguyên tố cùng nhau với v.

a. Vẽ đồ thị trên mặt phẳng. Đếm số cạnh của đồ thị.

b. Xác định nửa bậc vào và nửa bậc ra của các đỉnh.

c. Chứng minh đồ thị là liên thông yếu nhưng không liên thông mạnh.

d. Tìm độ dài của đường đi đơn dài nhất trong đồ thị.

3. a. Cho một mạng máy tính gồm 41 máy, mỗi máy được nối ít nhất với 20 máy khác. Chứng minh mạng là liên thông.

b. Tổng quát hóa kết quả câu a: chứng minh rằng một đồ thị vô hướng n đỉnh (n > 1) có bậc của mỗi đỉnh ít nhất bằng (n − 1)/2 là một đồ thị liên thông.

4. Trong dịp Tết, 9 người bạn hứa với nhau là mỗi người phải gửi thiếp chúc Tết tới 5 người trong số họ. Chứng minh rằng ít nhất có một người không gửi thiếp chúc Tết cho người đã gửi thiếp chúc Tết mình.

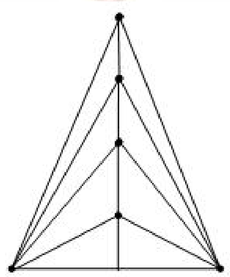
5. Xét đồ thị có tập đỉnh là các số nguyên từ 1 đến n. Hai đỉnh i ≠ j được nối với nhau bằng một cạnh nếu i + j là một số chẵn.

a. Chứng minh đồ thị không liên thông và xác định các thành phần liên thông của đồ thị này.

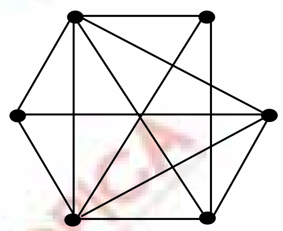
b. Nếu thay điều kiện i + j là số chẵn bằng điều kiện i + j là số lẻ thì kết quả sẽ như thế nào?

6. Xét xem hai đồ thị G, G’ trong các hình vẽ dưới đây có đẳng cấu không?

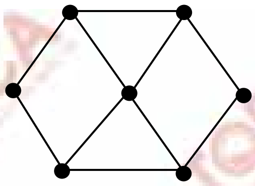
a.



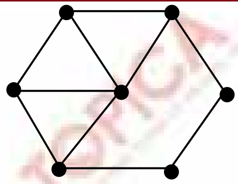
Đồ thị G



Đồ thị G′

b. 

Đồ thị G



Đồ thị G′

7. Cho một đồ thị vô hướng có bốn đỉnh có bậc 3, các đỉnh còn lại có bậc 2.

a. Chứng minh rằng số cạnh của đồ thị lớn hơn số đỉnh của nó 2 đơn vị.

b. Xây dựng hai đơn đồ thị liên thông, không đẳng cấu với nhau, có 6 đỉnh thỏa mãn điều kiện đã nêu.

8. Chứng minh rằng đỉnh u của một đơn đồ thị vô hướng liên thông (có số đỉnh > 2) là đỉnh khớp khi và chỉ khi tìm được hai đỉnh v, w khác nhau và khác u sao cho mọi đường đi nối v, w đều qua u.

9. Chứng minh rằng trong đơn đồ thị vô hướng có số đỉnh > 1 đỉnh luôn tìm được 2 đỉnh không phải đỉnh khớp.

10. Chứng minh rằng nếu một đơn đồ thị vô hướng m cạnh, n đỉnh là liên thông thì nó thỏa mãn bất đẳng thức:

$$m \geq n - 1$$

(điều kiện cần của đơn đồ thị vô hướng liên thông).

11. Cho đơn đồ thị vô hướng, m cạnh, n đỉnh, k thành phần liên thông. Chứng minh bất đẳng thức

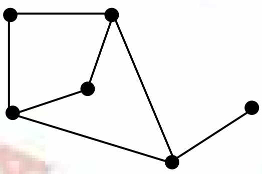
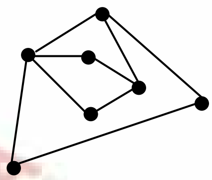
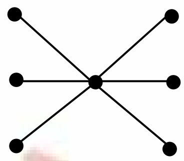
$$n - k \leq m \leq (n - k)(n - k + 1)/2.$$

12. Chứng minh rằng nếu một đơn đồ thị vô hướng m cạnh, n đỉnh thỏa mãn bất đẳng thức:

$$m > (n - 1)(n - 2)/2$$

thì đồ thị là liên thông (điều kiện đủ của đơn đồ thị vô hướng liên thông).

13. Thử lại các đồ thị dưới đây là hai phía và tìm phân hoạch hai lớp tương ứng trên tập đỉnh sao cho cạnh nối chỉ có giữa hai đỉnh thuộc hai lớp khác nhau.

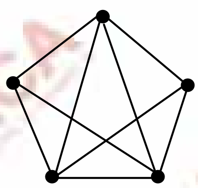
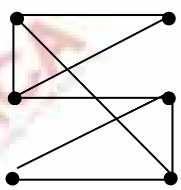
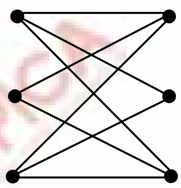
14. Cho G là đồ thị hai phía với n đỉnh và m cạnh.

a. Chứng minh rằng $m \leq n^2/4$.

b. Gọi K và k là bậc lớn nhất và nhỏ nhất của các đỉnh của G. Chứng minh rằng:

$$k \leq 2m/n \leq K.$$

15. Tìm biểu diễn phẳng của các đồ thị dưới đây:

16. Chứng minh các đồ thị dưới đây là không phẳng bằng cách chỉ ra một đồ thị con là không phẳng:

17. Xét đồ thị đầy đủ n đỉnh K_n ($n > 2$).

a. Chứng minh nó là đồ thị Hamilton.

b. Tìm điều kiện để K_n là đồ thị Euler.

18. Tìm điều kiện cho m, n để đồ thị hai phía đầy đủ $K_{m,\,n}$ ($m, n > 1$) là:

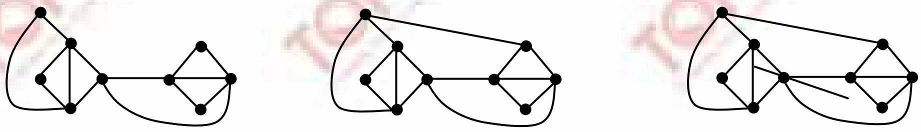
a. Euler.

b. Nửa Euler.

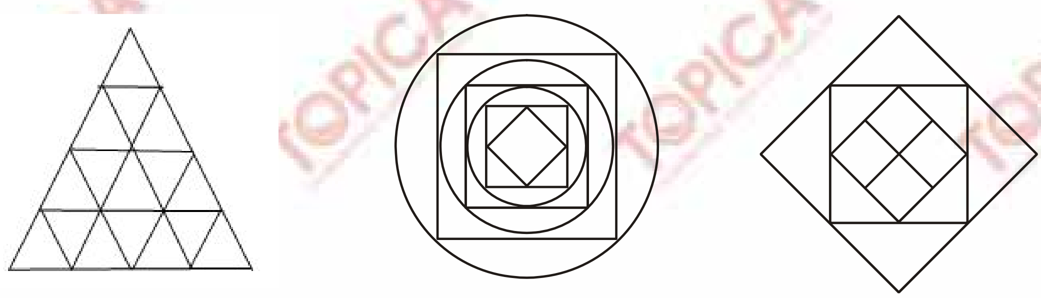
c. Hamilton.

d. Nửa Hamilton.

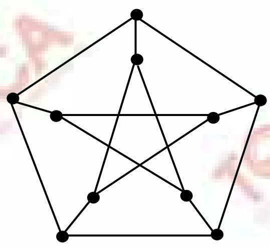
19. Xét tính Euler (không Euler, nửa Euler, Euler) của các đồ thị dưới đây, trong trường hợp đồ thị là Euler (nửa Euler), dùng thuật toán Fleury để tìm một chu trình Euler (đường đi Euler) trên đồ thị tương ứng:



20. Vẽ một nét các hình dưới đây (nếu thỏa mãn điều kiện vẽ được một nét):



21. Cho đồ thị Petersen:



a. Dùng quay lui để kiểm tra tính Hamilton của đồ thị đã cho. Nếu đồ thị là Hamilton (nửa Hamilton) thì chỉ ra một chu trình Hamilton (đường đi Hamilton) tìm được.

b. Hỏi như trên với đồ thị nhận được từ đồ thị Petersen bằng cách bỏ đi một đỉnh tùy ý (và các cạnh có liên quan).

Các bài tập dưới đây là các bài tập lập trình.

22. Lập trình nhập một đồ thị dưới dạng ma trận kề (từ file hoặc từ bàn phím), sau đó kiểm tra nó có phải là hai phía không? Nếu đúng, hãy đưa ra màn hình phân hoạch tương ứng tìm được.
Dùng chương trình thử lại kết quả bài tập 13.

23. Lập trình nhập một đồ thị dưới dạng danh sách kề (từ file hoặc từ bàn phím), sau đó kiểm tra tính Euler của đồ thị đã cho. Nếu nó là Euler (nửa Euler) thì đưa ra màn hình chu trình Euler (đường đi Euler) tìm được.
Dùng chương trình thử lại kết quả bài tập 19.

24. Lập trình nhập một đồ thị dưới dạng ma trận kề (từ file hoặc từ bàn phím), sau đó kiểm tra tính Hamilton của đồ thị đã cho. Nếu nó là Hamilton (nửa Hamilton) thì đưa ra màn hình chu trình Hamilton (đường đi Hamilton) tìm được.
Dùng chương trình thử lại kết quả bài tập 21.

CÂU HỎI THƯỜNG GẶP

1. Phân biệt sự khác nhau giữa đồ thị vô hướng và đồ thị có hướng?
2. Khái niệm bậc của đỉnh trong đồ thị vô hướng và có hướng có khác nhau không?
3. Phân biệt sự khác nhau giữa đồ thị và đa đồ thi?