ÔN TẬP HỌC KỲ II

I. CÁC KHÁI NIỆM CƠ BẢN VỀ ĐỒ THỊ

1. Khái niệm đồ thị

Đồ thị là một mô hình biểu diễn một tập các đối tượng và mối quan hệ hai ngôi giữa các đối tượng

Graph = Objects + Connections

Ký hiệu: G = (V, E).

Trong đó V (vertices) tập các đỉnh và E (edges) là tập các cạnh.

2. Một số khái niệm khác

a. *Cạnh liên thuộc, đỉnh kề, bậc*

Xét đồ thị G=(V, E), xét cạnh e ∈ E. Nếu e = (u, v) thì ta nói đỉnh u và đỉnh v kế nhau và cạnh e là cạnh liên thuộc của đỉnh u và đỉnh v.

Với một đỉnh v trong đồ thị vô hướng thì bậc (degree) của v ký hiệc deg(v) là số cạnh liên thuộc của v và cũng là số đỉnh kề với v.

Định lý 1: Trong đồ thị vô hướng G=(V,E) thì $\sum_{v \in V} \deg(v) = 2|E|$

Hệ quả: Trong đồ thị vô hướng tổng các đỉnh bậc lẻ là số chẳn.

Định lý 2: Trong đồ thị có hướng, tổng tất cả các bán bậc vào của đỉnh bằng tổng tất cả các bán bậc ra của đỉnh và bằng số cung.

$$\sum_{v \in V} \deg^{+}(v) = \sum_{v \in V} \deg^{-}(v) = |E|$$

b. *Đường đi và chu trình*

Một dãy các đỉnh P=<p_0, $p_{1,\ldots,}p_k$> sao cho (p_{i-1}, p_i) ∈ E và 1≤i≤k được gọi là đường đi. Đường đi có k+1 đỉnh và k cạnh.

Một đường đi đơn là đường đi mà tất cả các đỉnh là phân biệt.

Nếu đường đi đơn mà có p_0 trùng với p_k thì đường đi đó được gọi là chu trình đơn (k≤3).

c. Một số khái niệm khác

Đồ thị vô hướng liên thông là đồ thị mà hai đỉnh bất kỳ của đồ thị đều tồn tại đường đi.

Đối với đồ thị có hướng có hai khái niệm liên thông:

+ Liên thông mạnh: hai đỉnh bất kỳ của đồ thị đều tồn tại đường đi

+ liên thông yếu: phiên bản vô hướng của nó là liên thông.

+ Đồ thị vô hướng đầy đủ là đồ thị mà mọi cặp đỉnh của nó đều có cạnh.

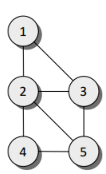
+ Đồ thị hai phía là đồ thị mà tập đỉnh của nó được chia thành hai tập X và Y. Nếu |X| = m và |Y| = n và mọi cặp đỉnh (x,y) trong đó x ∈ X và y ∈ Y đều có cạnh nối thì đồ thị hai phía đó được gọi là đồ thị hai phía đầy đủ, ký hiệu $K_{m,n}$.

+ Cây là một đồ thị vô hướng liên thông không có chu trình.

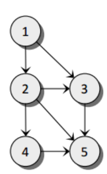
II.BIỂU DIỄN ĐỒ THỊ

1. Ma trận kề

Với đồ thị G=(V, E) và |V| = n, ta có thể đánh số thứ tự các đỉnh từ 1 đến n. G có thể biểu diễn bằng ma trận vuông A = $\{a_{i,j}\}_{nxn}$. Trong đó $a_{ij} = \begin{cases} 1 & \text{nếu } (i,j) \in E \\ 0 & \text{nếu } (i,j) \notin E \end{cases}$



$$\begin{pmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{pmatrix}$$



$$\begin{pmatrix} 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Nhận xét:

+ Nếu G vô hướng thì $a_{ij} = a_{ji}$ nên ma trận A là ma trận đối xứng qua đường chéo chính;

+ G có hướng thì deg(i) = số các số trên dòng i hay cột i.

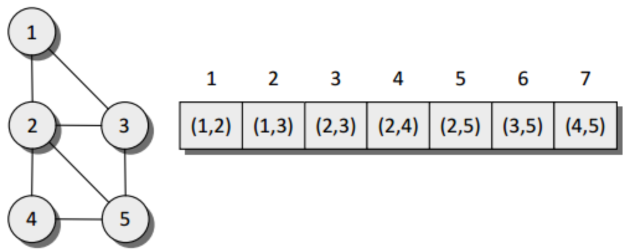
+ G có hướng thì $deg^+(i) = deg^-(i)$ = số các số trên dòng i.

+ Ưu điểm: dễ cài đặt và thao tác dễ thao tác.

+ Nhược điểm: tốn bộ nhớ, và một số thao tác không cần thiết.

2. Danh sách cạnh

Đồ thị G=(V,E) có n đỉnh và m cạnh, ta có thể liệt kê danh sách các cạnh của nó. Mỗi cạnh là một cặp số (x, y). Ví dụ như hình vẽ sau:



Ưu điểm:

+ Với đồ thị thưa cách lưu trữ này tiết kiệm được bộ nhớ;

+ Việc duyệt các cạnh dễ dàng.

Nhược điểm:

+ Duyệt đỉnh phải duyệt hết tất cả các cạnh;

+ Kiểm tra đỉnh kề cũng duyệt hết tất cả các cạnh.

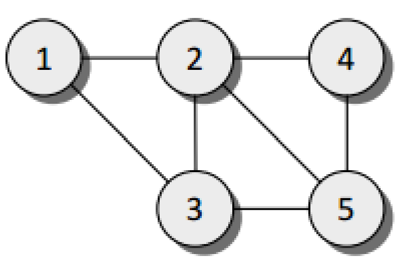
3. Danh sách kề

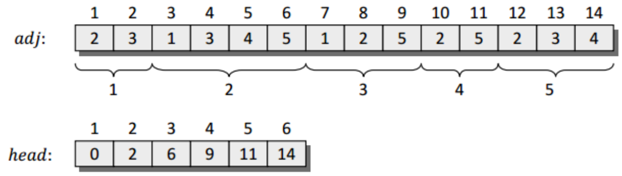
Với đồ thị G=(V,E) có n đỉnh và m cung, có hai cách cài đặt:

Forward star: với đỉnh u có một danh sách adj(u) chứa các đỉnh v nối từ u.

Reverse srar: với đỉnh v có một danh sách adj(v) chứa các đỉnh u nối đến v.

Biểu diễn danh sách kề bằng mảng một chiều:





Mỗi đỉnh trong H có một danh sách các đỉnh kề với nó nằm trong A.

Các đỉnh kề với u trong H được lưu trữ trong A từ A[H[u]+1] đến A[H[u+1]]

Ưu điểm:

+ Dễ dàng duyệt các đỉnh và các cạnh. **Nhược điểm:**

+ Để kiểm tra (u, v) có phải là cạnh hay không, cần phải duyệt toàn bộ (yếu hơn ma trận kề)

III. CÂY, CÂY KHUNG

a. Cây

- Cây là đồ thị liên thông không có chu trình

- Một số tính chất của cây

  + Với một cây có n đỉnh sẽ có đúng n-1 cạnh

  + Giữa hai đỉnh bất kỳ của cây có đúng một đường đi nối chúng

  + Nếu thêm bất cứ một cạnh nào ở ngoài vào cây thì cây sẽ xuất hiện chu trình.

  + Mỗi cạnh của cây là một cầu

b. Cây khung

Cho đồ thị G=<V, E>.

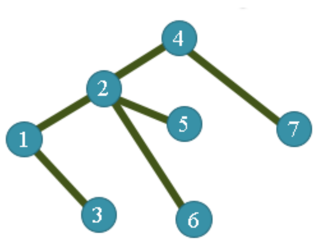
Cây khung của G là cây có tập đỉnh là tậ V của G và các cạnh ccủa cây thuộc E.

Ví dụ: Cho đồ thị G

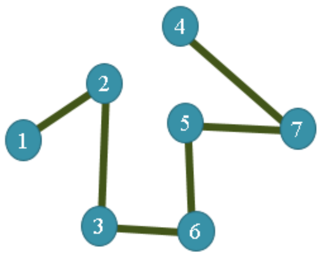
Ta có thể dùng phép duyệt đồ thị để tìm cây khung

Kết quả

Cây khung duyệt theo chiều rộng -BFS



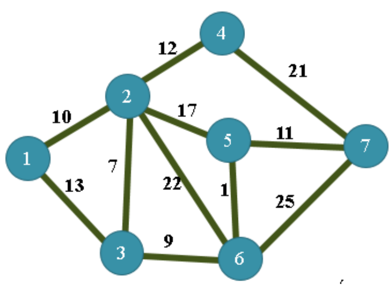
Cây khung duyệt theo chiều sau - DFS



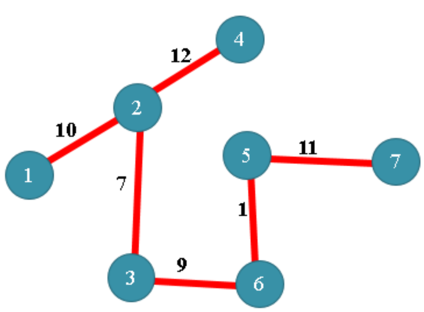
c. Bài toán tìm cây khung nhỏ nhất

Cho đồ thị vô hướng G=<V, E> có trọng số. Hãy tìm cây khung T có tổng trọng số của các cạnh là nhỏ nhất.

Ví dụ: Cho đồ thị sau



Cây khung nhỏ nhất tìm được có tổng trọng số là 50 như sau:



d. Thuật toán Kruskal

Ý tưởng: Xây dựng cây khung T bằng cách chọn từng cạnh từ E đưa vào T cho đến khi đủ n-1 cạnh. Cạnh được chọn phải có trọng số bé nhất và khi đưa nó vào T thì không được tạo chu trình.

Thuật toán:

B1: Sắp xếp các cạnh trong E theo thứ tự tăng dần của trọng số

B2: Khởi tạo T=Φ

B3: Chọn cạnh bé nhất đưa vào T

B4: Lặp cho đến khi T đủ n-1 cạnh

+ Chọn cạnh *e* bé nhất từ dãy đã sắp xếp

+ Nếu T+{*e*} không có chu trình thì đưa *e* vào T.

PHẦN II: BÀI TẬP

+ Quay lui

+ Quy hoạch động

+ Đồ thị