

**Định nghĩa:**  
Một **đồ thị** GGG là một cặp G=(V,E)G=(V,E)G=(V,E) với VVV là tập các đỉnh (vertices) và EEE là tập các cạnh (edges). Mỗi cạnh nối hai đỉnh (có thể có hướng hoặc không).

**Thành phần:**

* Tập đỉnh V={v1,v2,…,vn}V=\{v\_1,v\_2,\dots,v\_n\}V={v1​,v2​,…,vn​}.
* Tập cạnh EEE. Với đồ thị vô hướng, cạnh là tập 2-phan tử {u,v}\{u,v\}{u,v}. Với đồ thị có hướng, cung là cặp có thứ tự (u,v)(u,v)(u,v).

**Các loại đồ thị chính:**

* **Đồ thị vô hướng (undirected):** cạnh không có hướng.
* **Đồ thị có hướng (directed / digraph):** mỗi cạnh là cung có hướng.
* **Đồ thị có trọng số (weighted):** mỗi cạnh kèm trọng số/chi phí.
* **Đồ thị đơn (simple graph):** không có nhiều cung/cạnh giữa cùng hai đỉnh, không có vòng tự (loop).
* **Đa đồ thị (multigraph):** cho phép nhiều cạnh giữa cùng hai đỉnh.
* **Đồ thị có vòng (pseudograph):** cho phép vòng tự (edge from a vertex to itself).
* **Đồ thị liên thông / không liên thông, cây, đồ thị đầy đủ (complete), bipartite, v.v.**

**Ví dụ minh họa (vô hướng, 4 đỉnh):**  
Đỉnh: {1,2,3,4}, Cạnh: { {1,2}, {1,3}, {3,4} }.



**Ma trận kề (adjacency matrix)** của đồ thị có nnn đỉnh là ma trận A=[aij]A=[a\_{ij}]A=[aij​] kích thước n×nn\times nn×n với:

* **Đồ thị vô hướng, không trọng số (0/1):**

A black background with white text

AI-generated content may be incorrect.

* **Đồ thị có hướng:**

A black and white text

AI-generated content may be incorrect.

Ma trận có thể **không đối xứng**.

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A black and white text on a black background

AI-generated content may be incorrect.

A black background with white text

AI-generated content may be incorrect.

**8) Mối quan hệ giữa ma trận kề và đỉnh cô lập (isolated vertex)**

* **Đỉnh cô lập** là đỉnh không có cạnh nào nối tới hoặc ra khỏi nó.
* Trong ma trận kề (cả hướng và vô hướng), **đỉnh i cô lập ⇔ tất cả phần tử ở hàng i và cột i đều bằng 0**.
  + Với đồ thị vô hướng, hàng i = cột i, nên chỉ cần hàng i toàn 0.
  + Bậc (degree) = tổng hàng i = 0.

**Ví dụ:** nếu hàng và cột thứ 3 toàn 0 thì đỉnh 3 là đỉnh cô lập.

**9) So sánh giống nhau và khác nhau giữa đồ thị vô hướng và có hướng**

**Giống nhau:**

* Cả hai đều gồm tập đỉnh VVV và tập cạnh/cung EEE.
* Cả hai có thể biểu diễn bằng **ma trận kề** và **danh sách kề**.
* Nhiều khái niệm như đường đi, chu trình, thành phần liên thông (khái niệm khác nhau với có hướng) vẫn tương tự.

**Khác nhau (điểm chính):**

* **Hướng cạnh:** vô hướng: cạnh không có chiều; có hướng: cung có chiều → thứ tự các đỉnh quan trọng.
* **Ma trận kề:** vô hướng → ma trận **đối xứng**; có hướng → **không đối xứng** nói chung.
* **Độ (degree):** vô hướng có một bậc (degree); có hướng có **bậc vào** và **bậc ra**.
* **Liên thông:** vô hướng: “connected” (một loại). Có hướng: phân biệt **weakly connected** (bỏ hướng) và **strongly connected** (có đường đi theo hướng giữa mọi cặp).
* **Thuật toán/ứng dụng:** nhiều bài toán cho đồ thị có hướng (topological sort, phân lớp chu trình, mạng chảy) khác với đồ thị vô hướng (spanning tree, matching, v.v.).
* **Kiểm tra cạnh tồn tại:** cả hai cùng O(1) với ma trận; nhưng với danh sách kề, vô hướng lưu cả hai chiều hoặc chỉ lưu một chiều tùy cách.

**10) Vai trò và ứng dụng của đồ thị trong thực tế (vài ví dụ minh họa)**

Đồ thị là mô hình rất phổ biến; ứng dụng trải rộng nhiều lĩnh vực:

1. **Mạng xã hội:** đỉnh = người, cạnh = quan hệ bạn bè / follow. Dùng để tìm “influencer” (đỉnh có degree lớn), cộng đồng (community detection).
   * Ma trận kề cho biết ai kết nối với ai; phân tích bậc (degree) cho trung tâm mạng.
2. **Mạng giao thông / bản đồ:** đỉnh = giao lộ, cạnh = đường (trọng số = thời gian/chi phí).
   * Giải bài toán đường đi ngắn nhất (Dijkstra), luồng tối ưu (max-flow) cho lưu lượng.
3. **Mạng máy tính / routing:** nút = router, cạnh = link; xác định đường truyền, phân tích độ tin cậy, redundancy.
4. **Phân tích phụ thuộc / build (topological sort):** biểu diễn quan hệ phụ thuộc (task A phải xong trước B) bằng đồ thị có hướng, dùng topological sort để sắp xếp thực hiện.
5. **Sinh học:** mạng tương tác protein, mạng gen.
6. **Hóa học:** phân tử coi như đồ thị (đỉnh = nguyên tử, cạnh = liên kết) để phân tích cấu trúc.
7. **Search & ranking (PageRank):** biểu diễn web bằng đồ thị có hướng (link từ trang i → j). PageRank dùng ma trận chuyển tiếp dựa trên ma trận kề.
8. **Xử lý hình ảnh, tách vùng (segmentation):** sử dụng đồ thị để tách vùng ảnh.