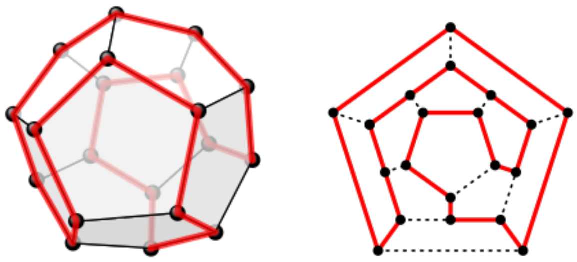
# BÀI 7
# ĐƯỜNG ĐI VÀ CHU TRÌNH HAMILTON

## Mục tiêu

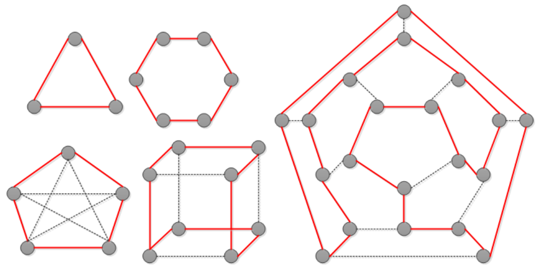
- Hiểu các khái niệm đường đi Hamilton, chu trình Hamilton, đồ thị Hamilton, đồ thị nửa Hamilton;
- Hiểu ý tưởng và cài đặt được thuật toán để tìm đường đi/chu trình Hamilton;
- Vận dụng giải các bài toán liên quan.

## 1. Một số khái niệm



*Hình 1. Trò chơi "icosian" do nhà toán học Hamilton đặt ra năm 1856*

> **Bài toán:** Xuất phát từ một đỉnh của khối thập nhị diện đều *(khối cầu 12 mặt, mỗi mặt là một ngũ giác)* hãy đi dọc theo các cạnh của khối đó sao cho đi qua tất cả các đỉnh khác, mỗi đỉnh đúng một lần sau đó quay về đỉnh xuất phát.



*Hình 2. Một số đồ thị Hamilton*

**Đường đi Hamilton** *(Hamiltonian path):* Một đường đi trong đồ thị $G = (V, E)$ được gọi là đường đi Hamilton nếu **đi qua tất cả các đỉnh** của đồ thị, **mỗi đỉnh đúng một lần**.

**Chu trình Hamilton** *(Hamiltonian cycle):* là đường đi Hamilton có đỉnh đầu và đỉnh cuối trùng nhau.

**Đồ thị Hamilton** *(Hamiltonian graph):* là đồ thị **chứa chu trình Hamilton**.

**Đồ thị nửa Hamilton** *(semi-Hamiltonian graph):* là đồ thị **chứa đường đi Hamilton**.

Khác với chu trình Euler, cho đến nay:

- **Chưa có điều kiện cần và đủ** để kiểm tra xem một đồ thị có chứa chu trình Hamilton không, các định lý bên dưới chỉ là **điều kiện đủ** để một đồ thị có chu trình Hamilton;
- Chưa có thuật toán hiệu quả để tìm chu trình Hamilton.

**Định lý Dirac (1952)**: Một đơn đồ thị vô hướng $G = (V, E)$ $n$ đỉnh ($n \geq 3$) có chu trình Hamilton nếu **mọi đỉnh đều có bậc lớn hơn hoặc bằng $\frac{n}{2}$**.

$$deg(v) \geq \left\lceil \frac{n}{2} \right\rceil \quad , \forall v \in V$$

**Định lý Ore (1960):** Một đơn đồ thị vô hướng $G = (V, E)$ $n$ đỉnh có chu trình Hamilton nếu **tổng bậc của mỗi cặp đỉnh không kề nhau** lớn hơn hoặc bằng $n$.

$$deg(u) + deg(v) \geq n \quad , \forall u, v \in V \text{ và } (u, v) \notin E$$

**Định lý Houiri (1960)**: Một đơn đồ thị có hướng $G = (V, E)$ $n$ đỉnh liên thông mạnh có chu trình Hamilton nếu **bậc tổng cộng của mọi đỉnh đều lớn hơn hoặc bằng $n$**.

$$deg^{+}(v) + deg^{-}(v) \geq n \quad , \forall v \in V$$

**Định lý Meyniel (1973)**: Một đơn đồ thị có hướng $G = (V, E)$ $n$ đỉnh liên thông mạnh có chu trình Hamilton nếu **tổng bậc tổng cộng của mỗi cặp đỉnh không kề nhau** lớn hơn hoặc bằng $2n - 1$.

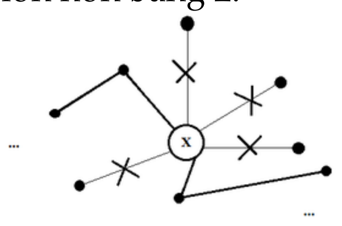
$$deg^{+}(u) + deg^{-}(u) + deg^{+}(v) + deg^{-}(v) \geq 2n - 1 \quad , \forall u, v \in V \text{ và } (u, v) \notin E$$

## 2. Thuật toán tìm chu trình Hamilton

### 2.1. Quy tắc tìm chu trình Hamilton

Hiện nay, dù chưa có thuật toán tổng quát để tìm chu trình Hamilton nhưng có một số quy tắc được sử dụng trong quá trình xây dựng chu trình Hamilton:

- **Quy tắc 1**: Chọn hết tất cả các cạnh liên thuộc với bậc 2;
- **Quy tắc 2**: Không cho tạo ra chu trình ít hơn n cạnh;
- **Quy tắc 3**: Khi đã chọn 2 cạnh liên thuộc với đỉnh thì xóa tất các các cạnh còn lại liên thuộc với $x$ (*vì không đi qua đỉnh u nữa*);
- **Quy tắc 4**: Khi áp dụng quy tắc 3, đảm bảo duy trì tính liên thông và bậc của các đỉnh còn lại luôn lớn hơn bằng 2.




*Hình 3. Minh họa quy tắc 3 khi xây dựng chu trình Hamilton*

## 2.2. Thuật toán vét cạn tìm chu trình Hamilton

**Cài đặt:**

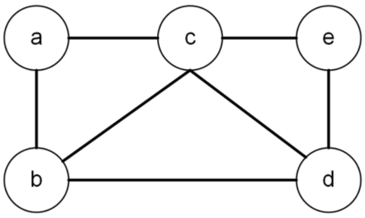
**Input**: đồ thị $G(V, E)$

**Output**: một chu trình Hamilton (nếu có)

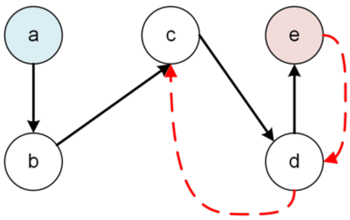
```
FindHamiltonianCycle()
   visited ← bool[n]
   x ← int[n+1]   //Mảng lưu n+1 đỉnh trong chu trình Hamilton
   x[0] ← x0      //Chọn đỉnh bắt đầu
   visited[x0] ← true

   //Tìm thành phần nghiệm tiếp theo
   hasHamiltonCycle ← FindVertex(1, x, visited)
   IF (!hasHamiltonCycle)
      RETURN NULL
   x[n] ← x[0]
   RETURN x


//Tìm đỉnh thứ i trong chu trình Hamilton
FindVertex(i, x, visited)
   //Chọn đủ n đỉnh
   IF (i==n)
      //Kiểm tra có cạnh nối đỉnh x[n-1] với x[0]
      RETURN x[i-1] IN adjList[x[0]]

   FOREACH v IN adjList[x[i-1]]
      IF (!visited[v])
         x[i] ← v            //Lựa chọn v làm đỉnh thứ i
         visited[v] ← true   //Đánh dấu đỉnh v
         //Tìm thành phần (i+1), nếu tìm được lời giải thì
         //không xét các đỉnh v còn lại
         IF (FindVertex(i+1, x, visited))
            RETURN true
         visited[v] ← false  //Không chọn v làm đỉnh thứ i
   RETURN false
```

**Độ phức tạp**: **O**(|V|!)

**Ví dụ:** Cho đồ thị như hình vẽ. Hãy tìm một chu trình Hamilton *(nếu có)*

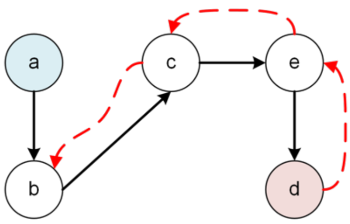



- Bắt đầu từ đỉnh a, lần lượt đi theo các đỉnh kề *(tương tự thuật toán DFS)*, tìm được một đường đi **{a, b, c, d, e}** đi qua tất cả các đỉnh nhưng từ đỉnh kết thúc *(đỉnh e)* không có cạnh nối với đỉnh bắt đầu *(đỉnh a)* nên không tạo thành chu trình Hamilton. Quay lui về đỉnh d để tìm đỉnh hướng đi khác nhưng không có. Tiếp tục quay lui về đỉnh c (*Hình 4*);



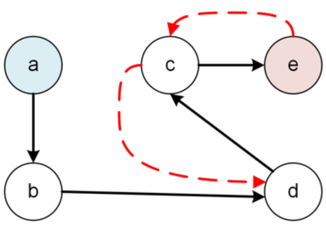

*Hình 4. Minh hoạ xây dựng chu trình Hamilton (1)*

- Tại đỉnh c, tìm được đường đi khác **{a, b, c, e, d}** đi qua tất cả các đỉnh nhưng từ đỉnh kết thúc *(đỉnh d)* không có cạnh nối với đỉnh bắt đầu *(đỉnh a)* nên không tạo thành chu trình Hamilton. Lần lượt quay lui về đỉnh e, đỉnh c rồi đỉnh b (*Hình 5*);



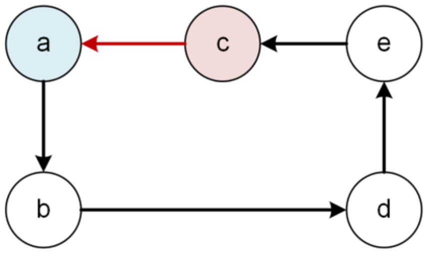

*Hình 5. Minh hoạ xây dựng chu trình Hamilton (2)*

- Tìm được đường đi **{a, b, d, c, e}** đi qua tất cả các đỉnh nhưng từ đỉnh kết thúc *(đỉnh e)* không có cạnh nối với đỉnh bắt đầu *(đỉnh a)* nên không tạo thành chu trình Hamilton. Lần lượt quay lui về đỉnh e, đỉnh c rồi đỉnh b (*Hình 6*);



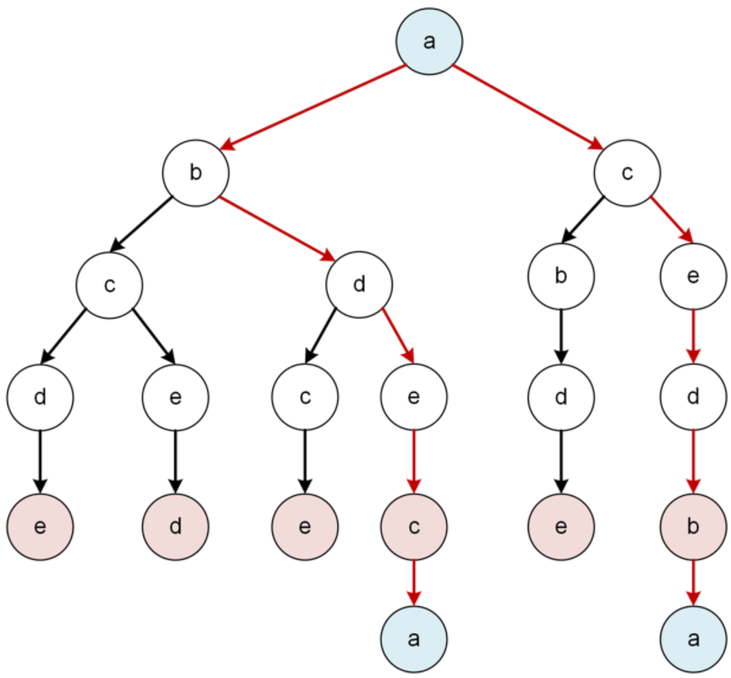

*Hình 6. Minh hoạ xây dựng chu trình Hamilton (3)*

- Tìm được đường đi **{a, b, d, e, c}** đi qua tất cả các đỉnh và từ đỉnh kết thúc *(đỉnh c)* có cạnh nối với đỉnh bắt đầu *(đỉnh a)* nên tạo thành chu trình Hamilton (*Hình 7*). Kết thúc thuật toán.

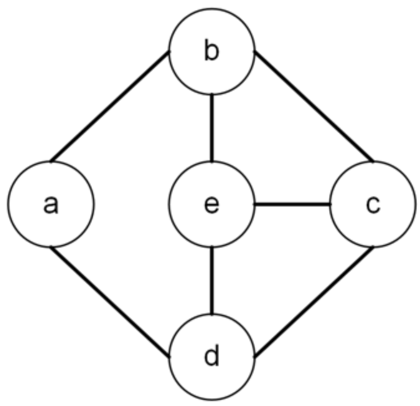



*Hình 7. Minh hoạ xây dựng chu trình Hamilton (4)*

- Quá trình xây dựng chu trình Hamilton có thể minh họa ngắn gọn bằng một cấu trúc cây như sau (*tìm tất cả chu trình Hamilton*):



**Ví dụ:** Cho đồ thị như hình vẽ. Hãy vẽ cây minh hoạ quá trình tìm tất cả chu trình Hamilton.




**--- HẾT ---**