# Bài 1. Con đường trọng yếu — CROAD

- Cho một đồ thị vô hướng liên thông. Cạnh $g$ được gọi là trọng yếu đối với cặp đỉnh $(x, y)$ nếu mọi đường đi từ $x$ đến $y$ đều đi qua $g$.
- Có $q$ truy vấn dạng $(e, u, v)$ với $e$ là một cạnh và $u, v$ là hai đỉnh bất kỳ của đồ thị, truy vấn này giả định xóa cạnh $e$ khỏi đồ thị sau đó đếm số cạnh trọng yếu đối với cặp đỉnh $(u, v)$.

# Subtask 1 (15 điểm): $n, m, q \leq 500$

- Với mỗi truy vấn ta xóa cạnh $e$ trong truy vấn đi, sau đó duyệt từng cạnh $e'$ ($e \neq e'$) trong đồ thị và tạm xóa cạnh $e'$ này ra khỏi đồ thị.
- Tiếp theo áp dụng thuật toán BFS loang từ đỉnh $u$ xem có đến được đỉnh $v$ hay không, nếu không đến được thì cạnh $e$ là cạnh trọng yếu.
- Lưu ý kiểm tra xem khi loại cạnh $e$ đi thì đồ thị nếu không tồn tại đường đi từ $u$ đến $v$ thì kết quả là 0.
- Độ phức tạp theo cách trên là $O(q \times m^2)$.

# Subtask 2 (15 điểm): $n, m, q \leq 5000$

- Với mỗi truy vấn, sau khi loại cạnh $e$ ra khỏi đồ thị, các cạnh là trọng yếu chính là cầu của đồ thị.
- Sử dụng thuật toán tìm khớp cầu cải tiến từ thuật toán DFS để đếm xem có bao nhiêu cầu trên đường đi từ $u$ đến $v$. Độ phức tạp là $O(q \times m)$.

## Subtask 3 (20 điểm): $n, m, q \leq 200000$, $k = 1$

- Vì ta biết chắc chắn cạnh $e = 1$ bị xóa trong tất cả các truy vấn, do đó ta bỏ cạnh 1 ra khỏi đồ thị, sau đó tìm tất cả cầu trên đồ thị này.
- Bài toán đưa về với mỗi cặp đỉnh $(u, v)$ cần đếm xem có bao nhiêu cầu giữa hai đỉnh này.
- Ta có thể coi các đỉnh liên thông với nhau mà không đi qua bất kỳ cầu nào là một đỉnh của một cây với các cạnh là cầu của đồ thị trước. Như vậy bài toán chuyển thành tính khoảng cách giữa hai đỉnh trên cây.
- Sử dụng kỹ thuật LCA để tính được khoảng cách giữa hai đỉnh trên cây.
- Độ phức tạp là $O(q \log n)$.

## Subtask 4-5 (50 điểm): $n, m, q \leq 200000$

- Để đơn giản, giả sử ban đầu đồ thị không có cầu.
- Thực hiện thuật toán DFS trên đồ thị. Các cạnh được duyệt bởi DFS được gọi là cạnh cây DFS, các cạnh còn lại được gọi là cạnh ngược (back-edge). Cạnh ngược có tính chất: nó chỉ nối 2 đỉnh có quan hệ tổ tiên-con cháu trên cây DFS.
- Với mỗi cạnh DFS $x$, gọi $S(x)$ là tập các cạnh ngược có một đỉnh một trong hai đỉnh đầu mút của $x$.
- Xét 2 cạnh cây DFS $x$ và $y$. Nếu xóa $x$ thì $y$ trở thành cầu, nên nếu xóa cả $x$ và $y$ sẽ khiến đồ thị mất tính liên thông. Vì vậy, $S(x) = S(y)$.
- Khi $S(x) = S(y)$, xóa cả $x$ và $y$ sẽ khiến đồ thị chia làm 2 thành phần trong đó có 1 phần "nằm giữa" $x$ và $y$.

# Subtask 4-5 (50 điểm): $n, m, q \leq 200000$

- Chia các cạnh cây DFS vào các tập sao cho mọi cạnh thuộc cùng tập có $S()$ bằng nhau.
- Khi gặp truy vấn $k, u, v$ với $e$ là cạnh thứ $k$ trên đồ thị:
  - Trường hợp 1: $e$ là cạnh ngược.
    Mọi cạnh cây DFS $x$ có $S(x) = \{e\}$ sẽ trở thành cầu. Cần đếm xem trong số các cạnh $x$ như thế, có bao nhiêu cạnh nằm trên đường đi trên cây từ $u$ đến $v$ (kỹ thuật giống subtask 3 sử dụng LCA).
  - Trường hợp 2: $e$ là cạnh cây DFS. Dễ thấy tập các cạnh có $S()$ bằng $S(e)$ tạo thành một "đường đi" (có thể không liên tiếp) từ trên xuống dưới. Cần đếm xem có bao nhiêu cạnh $x$ mà $S(x) = S(k)$ và đúng một trong 2 đỉnh $u, v$ thuộc phần "nằm giữa" $x$ và $k$.