**So sánh ưu - nhược điểm:**

1. **Độ rõ ràng của code:**

* **Cách 1 (Dictionary):**
  + Ưu: Logic tuần tự dễ theo dõi, phù hợp với lập trình viên ở mọi cấp độ
  + Nhược: Code dài hơn do xử lý thủ công
* **Cách 2 (LINQ):**
  + Ưu: Ngắn gọn, tận dụng sức mạnh của LINQ trong C#
  + Nhược: Yêu cầu hiểu biết về LINQ, có thể khó đọc với người mới

1. **Hiệu năng:**

* **Cách 1 (Dictionary):**
  + Ưu: O(n) thời gian tính toán, không cần sắp xếp
  + Nhược: Có thể tốn bộ nhớ hơn khi số địa chỉ lớn
* **Cách 2 (LINQ):**
  + Ưu: Tối ưu cho các tập dữ liệu lớn nhờ LINQ
  + Nhược: Overhead của LINQ có thể làm chậm với dữ liệu nhỏ

1. **Khả năng mở rộng:**

* **Cách 1 (Dictionary):**
  + Ưu: Dễ thêm logic phức tạp (ví dụ: lọc theo điều kiện khác)
  + Nhược: Khó tích hợp với các hệ thống xử lý dữ liệu lớn
* **Cách 2 (LINQ):**
  + Ưu: Dễ mở rộng với Entity Framework hoặc các nguồn dữ liệu khác
  + Nhược: Khó tùy chỉnh logic gộp phức tạp mà không làm mất tính ngắn gọn

**Kết luận:**

* Sử dụng Cách 1 khi cần code dễ bảo trì và dữ liệu không quá lớn
* Sử dụng Cách 2 khi làm việc với hệ thống lớn hoặc cần tích hợp với database