1. **Trình bày lại khái niệm host address và network address đã nói trên lớp**

- IP Address là một số duy nhất được gán cho một thiết bị trong một mạng. Các thiết bị này có thể là một máy tính, router, smartphone,…vv. IP Address là một số 32 Bit và được chia thành 4 phần mỗi phần 8 Bit và ngăn cách nhau bởi dấu chấm (.).

- Một địa chỉ IP bao giờ cũng có hai phần là địa chỉ mạng ( Network Address ) và địa chỉ máy ( Host Address ):

+ Network address là một số duy nhất dùng để xác định một mạng. Mỗi máy tính trong một mạng bao giờ cũng có cùng một địa chỉ mạng.

+ Host address là một số duy nhất được gán cho một máy tính trong một mạng.

1. **Giải thích rõ hơn khái niệm little-endian và big-endian?**

little-endian và big-endian là hai phương thức khác nhau để lưu trữ dữ liệu dạng nhị phân.

Trong cơ chế lưu trữ little endian (xuất phát từ "little-end" nghĩa kết thúc nhỏ hơn), byte cuối cùng trong biểu diễn nhị phân trên sẽ được ghi trước.

Big endian (xuất phát từ "big-end") thì ngược lại, là cơ chế ghi dữ liệu theo thứ tự bình thường mà chúng ta vẫn dùng.

little endian hay big endian khác nhau ở cách sắp xếp các byte dữ liệu còn thứ tự từng bit trong byte thì giống nhau. Chúng cũng khác nhau khi cần lưu trữ dữ liệu có nhiều byte, những dữ liệu chỉ có 1 byte (ví dụ ký tự ASCII) thì không ảnh hưởng gì.

Little endian hay big endian chỉ khác nhau ở việc lưu trữ thứ tự các byte dữ liệu. Cả hai phương thức đều không làm ảnh hưởng đến tốc độ xử lý của CPU. Thế nên cả hai phương thức đều vẫn tồn tại song song.

Mỗi phương thức đều có những lợi thế nhất định. Với little endian, vì byte nhỏ nhất luôn nằm bên trái, nó sẽ cho phép chúng ta đọc dữ liệu với độ dài tuỳ ý. Nó sẽ rất thích hợp nếu chúng ta cần ép kiểu.

Nhưng nếu cũng trường hợp đó, mà sử dụng big endian, thì chúng ta sẽ phải dịch địa chỉ bộ nhớ hiện tại thêm 4 byte nữa mới có không gian để lưu trữ.

Nhưng big endian cũng có những lợi thế nhất định, với việc đọc dữ liệu byte lớn nhất trước, nó sẽ rất dễ dàng kiểm tra một số là âm hay dương, do byte chứa dấu được đọc đầu tiên.

1. **Giải thích rõ hơn lệnh bind(): tại sao TCP client không cần sử dụng?**

Ở Server phải dùng lệnh Bind của Socket để yêu cầu hệ điều hành cho phép chiếm dụng một cổng. Lệnh Bind yêu cầu cung cấp một object thuộc kiểu IPEndPoint làm tham số.

Bởi vì Client không cần chờ trên một port UDP định sẵn nên nó cũng chẳng cần dùng phương thức Bind(), thay vì vậy nó sẽ lấy một port ngẫu nhiên trên hệ thống khi dữ liệu được gửi và nó giữa port này để nhận dữ liệu trả về.