Nội dung giáo trình được chia thành bảy chương chính như sau:

Chương 1: Tổng quan về lập trình mạng;

Chương 2: Công cụ và môi trường lập trình mạng;

Chương 3: Lập trình Socket - TCP Client/Server;

Chương 4: Lập trình Socket - UDP Client/Server;

Chương 5: Mô hình lập trình bất đồng bộ - Ứng dụng mạng có GUI;

Chương 6: Mô hình Multi-Client Server - Broadcast và Multicast;

Chương 7: Một số Protocol thông dụng - Xây dựng ứng dụng với dịch vụ HTTP và Email.

**CHƯƠNG 1. TỔNG QUAN VỀ LẬP TRÌNH MẠNG**

Chương 1 tập trung vào việc cung cấp một cái nhìn tổng quan về lập trình mạng - một lĩnh vực quan trọng và đa dạng trong ngành Công nghệ thông tin. Bằng cách trải qua lịch sử và sự phát triển của lập trình mạng từ những ngày đầu tiên của Internet đến hiện tại, chương này đưa ra tổng quan về cách mạng máy tính đã tạo nên cơ sở hạ tầng cho việc kết nối mạng toàn cầu và phát triển của các giao thức mạng. Tiếp theo, chương giới thiệu các khái niệm cơ bản trong lập trình mạng, từ giao thức và socket đến IP address và DNS, giúp định nghĩa và hiểu rõ các thuật ngữ quan trọng trong lập trình mạng. Sau đó, đi sâu vào các mô hình lập trình mạng phổ biến như mô hình client-server, peer-to-peer, giải thích cách mỗi mô hình hoạt động và được áp dụng trong thực tế. Cuối cùng, chương này cung cấp một số ví dụ về các ứng dụng thực tế của lập trình mạng, từ truyền tải file đến trò chơi trực tuyến, giúp định hình rõ hơn cách mà lập trình mạng ảnh hưởng và được áp dụng trong cuộc sống hàng ngày của chúng ta. Tóm lại, Chương 1 là điểm khởi đầu lý tưởng cho việc khám phá và hiểu sâu hơn về lập trình mạng, và là một cơ sở quan trọng cho những chủ đề tiếp theo trong cuốn sách.

**1.1 Lịch sử và sự phát triển của lập trình mạng**

Lịch sử Internet bắt đầu với việc phát triển máy tính điện tử trong những năm 1950. Các khái niệm ban đầu về mạng diện rộng bắt nguồn từ một số phòng thí nghiệm khoa học máy tính ở Mỹ, Vương quốc Anh và Pháp. Bộ Quốc phòng Mỹ đã trao các hợp đồng sớm nhất là vào những năm 1960, bao gồm cả việc phát triển dự án ARPANET. Tin nhắn đầu tiên được gửi qua ARPANET vào năm 1969 từ phòng thí nghiệm tại University of California, Los Angeles (UCLA) đến nút mạng thứ hai tại Stanford Research Institute (SRI).

Đến những năm 1970, bộ giao thức Internet (TCP/IP) được phát triển và trở thành giao thức mạng chuẩn trên ARPANET. Đầu những năm 1980, NSF đã tài trợ cho việc thành lập các trung tâm siêu máy tính quốc gia tại một số trường đại học và cung cấp kết nối vào năm 1986 với dự án NSFNET, cũng tạo ra quyền truy cập mạng vào các siêu máy tính ở Hoa Kỳ từ các tổ chức nghiên cứu và giáo dục. Các nhà cung cấp dịch vụ Internet (ISPs) bắt đầu xuất hiện vào cuối những năm 1980. ARPANET đã ngừng hoạt động vào năm 1990.

Vào những năm 1980, nghiên cứu tại CERN ở Thụy Sĩ của nhà khoa học máy tính người Anh Tim Berners-Lee đã đưa ra World Wide Web, liên kết các tài liệu siêu văn bản vào một hệ thống thông tin, có thể truy cập từ bất kỳ nút nào trên mạng. Từ giữa những năm 1990, Internet đã có một tác động mang tính cách mạng đối với văn hóa, thương mại và công nghệ, bao gồm sự gia tăng của giao tiếp gần như ngay lập tức bằng thư điện tử, tin nhắn tức thời, cuộc gọi qua điện thoại Giao thức Internet (VoIP), tương tác hai chiều các cuộc gọi video và World Wide Web với các diễn đàn thảo luận, blog.

Lập trình mạng không ngừng chuyển động và tiến hóa, đáp ứng đòi hỏi của một thế giới liên kết mạnh mẽ và đầy thách thức. Trong hiện tại, chúng ta chứng kiến sự tăng trưởng về ứng dụng và dịch vụ trên nền tảng mạng, từ các ứng dụng di động cho đến các hệ thống trí tuệ nhân tạo và Internet of Things (IoT). Điều này đặt ra một loạt các thách thức và cơ hội cho những người làm việc trong lĩnh vực lập trình mạng.

Một xu hướng quan trọng hiện nay là lập trình SDN (Software-Defined Networking) và NFV (Network Functions Virtualization). SDN cho phép quản lý mạng linh hoạt hơn thông qua việc tách biệt lớp điều khiển (control plane) và lớp chuyển tiếp dữ liệu (data plane). Còn NFV, cho phép các chức năng mạng truyền thống được chuyển từ các thiết bị vật lý sang các phần mềm chạy trên các máy ảo. Trong tương lai, lập trình mạng sẽ chuyển hướng vào việc tối ưu hóa mạng cho các ứng dụng đòi hỏi cao về băng thông và độ trễ, như trải nghiệm thực tế ảo (VR) và thực tế tăng cường (AR). Đồng thời, với sự ra đời của 5G, lập trình mạng sẽ được đặt ra những thách thức mới trong việc quản lý, tối ưu hóa mạng để đáp ứng yêu cầu của các ứng dụng di động với tốc độ, độ trễ tối thiểu. Nhìn chung, xu hướng lập trình mạng trong tương lai sẽ tiếp tục tập trung vào việc làm cho mạng trở nên linh hoạt, tự động hóa, có khả năng mở rộng, đáp ứng nhanh chóng với sự biến động không ngừng của nhu cầu công nghệ trong thế giới ngày nay và mai sau.

**1.2 Khái niệm cơ bản trong lập trình mạng**

Lập trình mạng liên quan đến việc phát triển các ứng dụng, dịch vụ mạng, cũng như quản lý, thiết kế và triển khai các hệ thống mạng. Dưới đây là một số khái niệm cơ bản trong lập trình mạng:

**1.2.1 Địa chỉ IP (IP Addressing)**

Là một địa chỉ số dạng số duy nhất được gán cho mỗi thiết bị trên mạng để xác định và liên kết với địa chỉ đó trên internet hoặc mạng nội bộ. Địa chỉ IP hiện tại chủ yếu sử dụng hai phiên bản: IPv4 và IPv6.

IPv4 (Internet Protocol version 4) là phiên bản giao thức Internet Protocol được sử dụng rộng rãi nhất trên Internet. Được phát triển ban đầu vào những năm 1980, IPv4 là hệ thống định danh duy nhất cho mỗi thiết bị kết nối vào mạng Internet. Dưới đây là một số điểm cần biết về IPv4:

**(1) Địa chỉ IPv4**: Mỗi thiết bị trên Internet được gán một địa chỉ IPv4 duy nhất, bao gồm 32 bit, được biểu diễn dưới dạng bốn octet được phân cách bằng dấu chấm. Ví dụ: 192.168.1.1.

**(2) Phạm vi địa chỉ IPv4:** IPv4 hỗ trợ khoảng 4 tỷ địa chỉ IP khác nhau, nhưng do sự phát triển nhanh chóng của Internet, số lượng địa chỉ IPv4 đã sắp cạn.

**(3) Cấp phát địa chỉ IPv4:** Địa chỉ IPv4 được cấp phát bởi các tổ chức quản lý phân phối địa chỉ IP, như ARIN (American Registry for Internet Numbers) cho khu vực Bắc Mỹ.

**(4) Quản lý địa chỉ IP:** IPv4 sử dụng giao thức DHCP (Dynamic Host Configuration Protocol) để tự động cấp phát địa chỉ IP cho các thiết bị kết nối vào mạng.

**(5) Định dạng header:** Header của gói tin IPv4 chứa các trường thông tin như phiên bản IP, độ dài header, kiểu dịch vụ, độ dài của dữ liệu, checksum, địa chỉ nguồn và đích, v.v.

**(6) Các vấn đề:** IPv4 gặp phải các vấn đề như sự cạn kiệt địa chỉ IPv4, sự cần thiết phải sử dụng NAT (Network Address Translation) để quản lý số lượng địa chỉ IP, và các vấn đề về bảo mật.

**Bảng 1.1**Mô tả về địa chỉ IPv4

| **Address Class** | **Type** | **Range** | **Default Subnet Mask** | **Number of Networks** | **No of Hosts Per Network** | **Use** |
| --- | --- | --- | --- | --- | --- | --- |
| A | Public | 1.0.0.0 to 127.0.0.0 | 255.0.0.0 | 126 | 16,777,214 | Governments and Large Number of Hosts |
| Private | 10.0.0.0 to 10.255.255.255 | 255.0.0.0 |  |  |  |
| B | Public | 128.0.0.0 to 191.255.255.255 | 255.255.0.0 | 16,382 | 65,534 | Medium Companies |
| Private | 172.16.0.0 to 172.31.255.255 | 255.255.0.0 |  |  |  |
| C | Public | 192.0.0.0 to 223.255.255.255 | 255.255.255.0 | 2,097,150 | 254 | Small Companies and LANs |
| Private | 192.168.0.0 to 192.168.255.255 | 255.255.255.0 |  |  |  |
| D | N/A | 224.0.0.0 to 239.255.255.255 | Not Applicable | N/A | N/A | Reserved for Multicasting |
| E | N/A | 240.0.0.0 to 254.255.255.255 | Not Applicable | N/A | N/A | Experimental |
| Special | N/A | 127.0.0.1 to 127.255.255.255 | N/A | N/A | N/A | Loopback Testing |

Trong những năm gần đây, IPv6 (Internet Protocol version 6) đã được phát triển để thay thế IPv4 và giải quyết các vấn đề về cạn kiệt địa chỉ IP. Tuy nhiên, IPv4 vẫn được sử dụng rộng rãi và tiếp tục là một phần quan trọng của cơ sở hạ tầng Internet.

**1.2.2 Cổng (Port)**

Trong lập trình mạng, khái niệm về port đóng vai trò quan trọng như là các "cổng kết nối" cho các ứng dụng và dịch vụ trên internet. Mỗi thiết bị kết nối với internet thông qua một địa chỉ IP, và trong mỗi thiết bị này, có hàng ngàn cổng (ports) sẵn sàng để tiếp nhận và gửi dữ liệu. Cụ thể, mỗi dịch vụ hoặc ứng dụng trên một thiết bị có thể lắng nghe tại một cổng cụ thể.

Ví dụ, các trang web thường lắng nghe tại cổng 80 (cho HTTP) hoặc cổng 443 (cho HTTPS). Đồng thời, các dịch vụ email thường sử dụng cổng 25 (SMTP) để gửi email và cổng 110 (POP3) hoặc cổng 143 (IMAP) để nhận email. Khái niệm về port cho phép các ứng dụng và dịch vụ trên internet tương tác với nhau một cách cấu trúc và có tổ chức. Có thể nhiều ứng dụng chạy trên cùng một máy tính và sử dụng các cổng khác nhau, giúp chúng hoạt động song song mà không gây ra xung đột hoặc giao lẫn với nhau. Port không chỉ giới hạn trong việc truy cập internet; chúng cũng được sử dụng trong các mạng nội bộ của doanh nghiệp để kiểm soát việc truy cập các dịch vụ và tài nguyên mạng.

Qua việc quản lý và điều hướng dữ liệu thông qua các cổng, port đóng vai trò quan trọng trong việc duy trì an toàn, hiệu quả và cấu trúc hóa của mạng một cách có tổ chức.

**1.2.3 Socket**

Socket là điểm cuối trong liên kết truyền thông hai chiều (two-way communication) biểu diễn kết nối giữa Client - Server.

Các lớp Socket được ràng buộc với một cổng port (thể hiện là một con số cụ thể) để các tầng TCP (TCP Layer) có thể định danh ứng dụng mà dữ liệu sẽ được gửi tới.

Quá trình gửi/nhận từ Socket.

Socket có thể hiểu như là cái cửa/cổng.

Quá trình gửi đẩy dữ liệu ra khỏi cửa/cổng.

Quá trình gửi dựa vào cơ tầng Transport ở phía bên kia cửa/cổng để gửi dữ liệu đến cửa/cổng tại quá trình nhận.

Hai cửa/cổng có liên quan với nhau: một ở mỗi bên.

**Hình 1.1**Mô tả về Socket [1]

**1.2.4 Giao thức (Protocol)**

Trong lập trình mạng, giao thức là một tập hợp các quy tắc và chuẩn được thiết lập để thiết lập, duy trì và kết thúc các phiên kết nối giữa các thiết bị trong mạng. Giao thức đóng vai trò như một loại ngôn ngữ, cho phép các thiết bị khác nhau, chạy các hệ thống hoặc ứng dụng khác nhau, giao tiếp và trao đổi dữ liệu một cách hiệu quả. Các giao thức mạng định hình cách thông tin được chuyển giao qua internet hoặc các mạng nội bộ. Chúng xác định cách dữ liệu được đóng gói, gửi đi, và nhận, đồng thời xác định cách thiết bị trong mạng phản ứng với các tình huống nhất định. Có nhiều loại giao thức mạng khác nhau. Dưới đây là một số giao thức quan trọng:

TCP/IP (Transmission Control Protocol/Internet Protocol): Đây là giao thức cơ bản của internet. Nó đảm bảo rằng dữ liệu được chia thành gói tin, được gửi đi từ nguồn đến đích một cách đáng tin cậy.

HTTP/HTTPS (Hypertext Transfer Protocol/Secure): Sử dụng cho việc truy cập và truyền tải trang web. Giao thức HTTPS sử dụng mã hóa để đảm bảo an toàn cho việc truyền tải dữ liệu.

**Hình 1.2**Giao thức của con người và máy tính [1]

FTP (File Transfer Protocol): Được sử dụng để truyền tải file giữa máy tính của người dùng và máy chủ.

SMTP (Simple Mail Transfer Protocol): Được sử dụng để gửi email từ máy chủ email của người gửi đến máy chủ email của người nhận.

POP3/IMAP (Post Office Protocol 3/Internet Message Access Protocol): Sử dụng để nhận email từ máy chủ email của người nhận đến thiết bị của người nhận.

DNS (Domain Name System): Chuyển đổi tên miền thành địa chỉ IP, cho phép người dùng truy cập trang web bằng tên thay vì số IP.

Nhờ vào việc tuân thủ các giao thức này, internet và các mạng nội bộ có thể hoạt động một cách mạch lạc, đồng nhất và an toàn, giúp hàng triệu người trên khắp thế giới kết nối và truy cập thông tin một cách dễ dàng.

**1.2.5 Gói tin (Packet)**

Trong lập trình mạng, một gói tin là một đơn vị nhỏ của dữ liệu được truyền tải qua mạng. Mỗi khi bạn gửi hoặc nhận dữ liệu qua internet hoặc bất kỳ mạng nào khác, dữ liệu đó không được truyền đi dưới dạng một lượng lớn thông tin không kiểm soát. Thay vào đó, nó được chia thành các đơn vị nhỏ hơn, được gọi là gói tin. Mỗi gói tin bao gồm nhiều thông tin quan trọng:

- Địa chỉ Người Gửi và Người Nhận: Gói tin chứa địa chỉ của máy tính hoặc thiết bị gửi và địa chỉ của máy tính hoặc thiết bị nhận.

- Dữ Liệu: Phần lớn của gói tin chứa dữ liệu thực sự, ví dụ như trang web bạn đang yêu cầu hoặc phần nội dung của email bạn đang gửi.

- Thông Tin Điều Khiển: Gói tin cũng bao gồm các thông tin quản lý, chẳng hạn như thông tin kiểm soát lỗi, để đảm bảo việc gửi và nhận dữ liệu một cách đáng tin cậy.

Khi bạn gửi một yêu cầu truy cập trang web hoặc bất kỳ hành động nào liên quan đến mạng, dữ liệu của bạn được chia thành các gói tin. Điều này quan trọng vì nó cho phép dữ liệu được gửi qua mạng một cách hiệu quả và an toàn. Khi gói tin đến đích, chúng sẽ được hợp nhất lại để tạo thành dữ liệu ban đầu. Quá trình này được thực hiện tự động bởi các thiết bị mạng như router và máy chủ. Đối với lập trình viên mạng, việc hiểu và làm việc với các gói tin là rất quan trọng. Nó giúp họ theo dõi và kiểm soát việc truyền tải dữ liệu trong mạng, cũng như giúp họ xử lý các vấn đề liên quan đến lỗi và hiệu suất mạng.

**1.3 Mô hình lập trình mạng**

Mô hình lập trình mạng là một cách tiếp cận hệ thống để xây dựng, triển khai và quản lý các ứng dụng và dịch vụ mạng. Đây không chỉ là việc kết nối các thiết bị và truyền dẫn dữ liệu, mà còn là quá trình tối ưu hóa hiệu suất, bảo mật và khả năng mở rộng của mạng.

**1.3.1 Mô hình OSI**

Mô hình OSI (Open Systems Interconnection): Một trong những mô hình phổ biến nhất là mô hình OSI, một mô hình tiêu chuẩn được đề xuất bởi ISO (International Organization for Standardization) để mô tả cách thông tin chuyển giao qua mạng từ một thiết bị đến thiết bị khác. Mô hình này chia quá trình truyền tải dữ liệu thành 7 lớp, từ việc quyết định xác định đường đi cho dữ liệu đến việc chuyển đổi dữ liệu thành các tín hiệu vật lý để truyền qua cáp hoặc không dây.

**Hình 1.3 Mô hình OSI**

**1.3.2 Mô hình TCP/IP**

TCP/IP (Transmission Control Protocol/Internet Protocol): Đại diện cho chân lý của thế giới internet ngày nay. Đây không chỉ là một mô hình, mà là một hệ thống giao thức mạng chính thức được sử dụng trên hầu hết các mạng trên thế giới. Khác với OSI, mô hình TCP/IP chỉ có 4 lớp. TCP/IP là mô hình mà Internet hiện đại dựa trên, và nó giải quyết mọi thứ từ việc truyền dẫn dữ liệu vật lý đến các ứng dụng trên web.  Mô hình TCP/IP không chỉ giới hạn ở việc kết nối máy tính với internet. Nó còn được sử dụng trong nhiều mạng riêng tư và doanh nghiệp, là cơ sở của việc truyền dẫn dữ liệu qua các mạng từ xa và là cơ sở của nhiều dịch vụ internet hiện đại như trang web, email, trò chuyện trực tuyến và nhiều hơn nữa. Mô hình TCP/IP không chỉ là một phần của quá trình kết nối, mà là trái tim của cuộc cách mạng số hóa, đang kết nối và biến đổi cách chúng ta tương tác với thế giới thông tin. Mô hình này được chia thành hai phần chính:

- TCP (Transmission Control Protocol): TCP là "ngôn ngữ" mà các thiết bị trên internet sử dụng để trò chuyện với nhau. Nó đảm bảo rằng dữ liệu được gửi từ một điểm đến điểm khác một cách tin cậy và không bị mất trên đường đi. TCP chia dữ liệu thành các gói nhỏ, gửi chúng đến đích và đảm bảo rằng chúng đều đến đúng đích mục.

- IP (Internet Protocol): IP giống như địa chỉ của bạn trên internet. Nó định danh thiết bị của bạn và cho phép các gói dữ liệu biết được nơi nào chúng nên được gửi. Mỗi thiết bị trên internet có một địa chỉ IP riêng, giống như là địa chỉ nhà của bạn trong thế giới thực.

**Hình 1.4** So sánh mô hình OSI và TCP/IP

c) Mô hình Client/Server: Một mô hình lập trình mạng khác được sử dụng rộng rãi là mô hình Client-Server, trong đó các máy tính cá nhân hoặc thiết bị khách hàng (Clients) kết nối đến máy chủ (Server) để lấy thông tin hoặc dịch vụ. Mô hình này chia sự chịu trách nhiệm của mạng thành hai loại thiết bị chính, giúp tối ưu hóa việc chia sẻ tài nguyên và quản lý tải.

**Hình 1.5** Mô hình mạng Client - Server [2]

d) Mô hình Peer-to-Peer: Một mô hình lập trình mạng tiếp theo là mô hình Peer-to-Peer (P2P), trong đó các thiết bị trên mạng được xem như ngang hàng và có khả năng chia sẻ tài nguyên trực tiếp với nhau mà không thông qua một máy chủ trung tâm. Mô hình này thường được sử dụng trong việc chia sẻ tệp tin hoặc dịch vụ mà không cần sự trung gian của máy chủ.

**Hình 1.6** Mô hình Peer-to-Peer [3]

e) Cuối cùng, mô hình Software-Defined Networking (SDN) là một phương pháp tiếp cận hiện đại trong lập trình mạng. SDN tách biệt phần quản lý mạng (control plane) và phần chuyển đổi dữ liệu (data plane), cho phép quản trị viên mạng kiểm soát và quản lý mạng thông qua phần mềm một cách linh hoạt và tự động.

**Hình 1.7** Mô hình mạng được xác định bằng phần mềm [3]

**1.4 Ứng dụng và ví dụ thực tế**

Ứng dụng hàng ngày của mạng máy tính: Hầu như mọi người đều sử dụng mạng máy tính hàng ngày, từ việc xem video trực tuyến, nghe nhạc, đến việc gửi email hay mua sắm trực tuyến.

Dưới đây là một số ứng dụng và ví dụ thực tế của lập trình mạng:

(1) Ứng dụng truyền tải file: Đây là các ứng dụng cho phép người dùng truyền tải file giữa các thiết bị qua mạng. Ví dụ: Dropbox, Google Drive, WeTransfer.

(2) Ứng dụng trò chơi trực tuyến: Các ứng dụng trò chơi trực tuyến kết nối người chơi với nhau thông qua mạng để chơi cùng nhau. Ví dụ: League of Legends, Fortnite, PUBG.

(3) Ứng dụng trò chuyện và gọi video: Các ứng dụng này cho phép người dùng trò chuyện văn bản, gọi video và chia sẻ dữ liệu qua mạng. Ví dụ: Skype, Zoom, WhatsApp.

(4) Ứng dụng mạng xã hội: Các ứng dụng mạng xã hội kết nối người dùng với nhau để chia sẻ thông tin, hình ảnh, video và tương tác. Ví dụ: Facebook, Instagram, Twitter.

(5) Ứng dụng thương mại điện tử: Các ứng dụng thương mại điện tử cho phép người dùng mua và bán hàng qua mạng. Ví dụ: Amazon, eBay, Shopee.

(6) Ứng dụng ngân hàng trực tuyến: Các ứng dụng này cho phép người dùng thực hiện các giao dịch tài chính qua mạng, như chuyển tiền, thanh toán hóa đơn và kiểm tra tài khoản. Ví dụ: Internet Banking của các ngân hàng.

(7) Ứng dụng IoT (Internet of Things): Các ứng dụng IoT kết nối và điều khiển các thiết bị thông minh qua mạng, như đèn, máy lạnh, camera an ninh. Ví dụ: Nest, Philips Hue, SmartThings.

(8) Ứng dụng video streaming: Các ứng dụng này cho phép người dùng xem video trực tuyến thông qua mạng. Ví dụ: Netflix, YouTube, Disney+.

Những ứng dụng này đều sử dụng lập trình mạng để tạo, quản lý và truyền thông dữ liệu giữa các thiết bị và người dùng qua mạng máy tính.

**1.5 Tóm tắt và kết luận**

Chương 1 đã cung cấp một cái nhìn tổng quan về lịch sử, khái niệm cơ bản, mô hình lập trình và các ứng dụng thực tế của lập trình mạng. Từ việc tìm hiểu về sự phát triển của mạng máy tính đến những khái niệm cơ bản như giao thức, socket và IP address, chương này đã giúp chúng ta hiểu rõ hơn về cách mà lập trình mạng ảnh hưởng và được áp dụng trong cuộc sống hàng ngày của chúng ta. Qua việc tìm hiểu các mô hình lập trình mạng phổ biến như Client-Server và Peer-to-Peer, chúng ta đã có cái nhìn sâu hơn về cách mà thông tin được truyền tải và xử lý qua mạng. Cuối cùng, bằng cách khám phá các ví dụ về các ứng dụng thực tế của lập trình mạng như truyền tải file, trò chơi trực tuyến và mạng xã hội, chúng ta đã nhận thấy vai trò quan trọng của lập trình mạng trong việc tạo ra các dịch vụ và sản phẩm mới, cũng như trong việc hiểu biết và tương tác với thế giới số hóa hiện đại. Tóm lại, chương 1 đã là một bước khởi đầu quan trọng trong việc hiểu và khám phá lập trình mạng, và là nền tảng cho việc tiếp tục khám phá những chủ đề phức tạp hơn trong các chương tiếp theo của cuốn sách.