

## Contents

DANH MỤC HÌNH VẼ .....	3
DANH SÁCH BẢNG BIỂU .....	3
DANH SÁCH TỪ VIẾT TẮT .....	3
MỞ ĐẦU .....	4
PHẦN I: NGUYÊN LÝ HỆ ĐIỀU HÀNH .....	5
CHƯƠNG 1: CƠ SỞ LÝ THUYẾT.....	5
CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG .....	5
CHƯƠNG 3: TRIỂN KHAI VÀ ĐÁNH GIÁ KẾT QUẢ.....	5
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....	5
PHẦN II: LẬP TRÌNH MẠNG .....	6
CHƯƠNG 1: CƠ SỞ LÝ THUYẾT.....	6
1. IP Helper.....	6
• Tổng quát về IP Helper .....	6
• Mục đích của IP Helper .....	7
• Nơi áp dụng IP Helper .....	7
• Đối tượng phát triển .....	7
• Các yêu cầu về môi trường run-time .....	7
2. Windows Sockets 2 .....	7
• Mục đích.....	7
• Đối tượng phát triển .....	8
• Các yêu cầu về môi trường run-time .....	8
3. Sử dụng IP Helper .....	8
4. Lệnh netstat .....	9
• Cú pháp .....	10
• Tham chiếu.....	10
• Chú giải.....	10
CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG .....	11
CHƯƠNG 3: TRIỂN KHAI VÀ ĐÁNH GIÁ KẾT QUẢ.....	11
1. Cài đặt chương trình .....	11

2. Kết quả chạy chương trình.....	11
3. Đánh giá kết quả .....	11
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....	11

**DANH MỤC HÌNH VẼ**  
**DANH SÁCH BẢNG BIỂU**  
**DANH SÁCH TỪ VIẾT TẮT**

## MỞ ĐẦU

## PHẦN I: NGUYÊN LÝ HỆ ĐIỀU HÀNH

**TIÊU ĐỀ:** Tìm hiểu và mô phỏng các giải thuật lập lịch trên hệ thống thời gian thực

**CHƯƠNG 1: CƠ SỞ LÝ THUYẾT**

**CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG**

**CHƯƠNG 3: TRIỂN KHAI VÀ ĐÁNH GIÁ KẾT QUẢ**

**KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

## PHẦN II: LẬP TRÌNH MẠNG

### **TIÊU ĐỀ: Tìm hiểu về IP Helper và xây dựng chương trình my\_netstat**

Yêu cầu đề tài:

1. Tìm hiểu về IP Helper và chương trình netstat.
2. Xây dựng được chương trình my\_netstat để mô phỏng chương trình netstat.

### **CHƯƠNG 1: CƠ SỞ LÝ THUYẾT**

#### **1. IP Helper**

- Tổng quát về IP Helper
  - Trình trợ giúp giao thức Interenet (IP Helper) hỗ trợ quản trị mạng cho máy tính cục bộ bằng cách cho phép các ứng dụng truy xuất thông tin về cấu hình của mạng của máy tính cục bộ và sửa đổi cấu hình đó. IP Helper cũng cung cấp các cơ chế thông báo để đảm bảo rằng một ứng dụng được thông báo khi các khía cạnh nhất định của cấu hình mạng của máy tính cục bộ bị thay đổi.
  - Nhiều hàm của IP Helper truyền các tham số có cấu trúc đại diện cho các loại dữ liệu được liên kết với công nghệ MIB (Management Information Base – là một cấu trúc dữ liệu gồm các đối tượng được quản lý, được dùng cho các thiết bị chạy trên nền tảng TCP/IP). IP Helper sử dụng các cấu trúc này để thể hiện các thông tin mạng khác nhau, chẳng hạn như các mục đệm ARP (Address Resolution Protocol – giao thức phân giải địa chỉ). Mặc dù IP Helper API sử dụng các cấu trúc cấu trúc này nhưng IP Helper khác biệt so với MIB và SNMP (Simple Network Management Protocol – quản lý giao thức mạng đơn giản).
  - IP Helper cung cấp các khả năng áp dụng trong các lĩnh vực sau:
    - Lấy thông tin cấu hình mạng
    - Quản lý bộ điều hợp mạng
    - Quản lý giao diện
    - Quản lý các địa chỉ IP

- Sử dụng giao thức phân giải địa chỉ
  - Lấy thông tin về giao thức Internet và giao thức tin nhắn điều khiển Internet
  - Quản lý định tuyến
  - Nhận thông báo về các sự kiện mạng
  - Lấy thông tin về giao thức điều khiển truyền vận (TCP) và giao thức dữ liệu người dùng (UDP).
- Mục đích của IP Helper
    - IP Helper API cho phép truy xuất và sửa đổi cài đặt cấu hình mạng cho máy tính cục bộ.
  - Nơi áp dụng IP Helper
    - IP Helper API có thể được áp dụng trong bất kì môi trường điện toán nào có cấu hình mạng và cấu hình TCP/IP có thể lập trình được. Một số ứng dụng điển hình bao gồm các giao thức định tuyến IP và các chương trình SNMP (Simple Network Management Protocol – quản lý giao thức mạng đơn giản).
  - Đối tượng phát triển
    - IP Helper API được thiết kế để sử dụng bởi các nhà lập trình viên C/C++.
  - Các yêu cầu về môi trường run-time
    - IP Helper API có thể được sử dụng trên tất cả các nền tảng Windows. Không phải tất cả hệ điều hành đều hỗ trợ tất cả các hàm của IP Helper. Nếu một hàm của IP Helper được sử dụng trên một nền tảng không được hỗ trợ, lỗi ERROR\_NOT\_SUPPORTED sẽ được trả về.

## 2. Windows Sockets 2

- Mục đích
  - Windows Sockets 2 (Winsock) cho phép các nhà lập trình viên tạo ra các ứng dụng internet, intranet hoặc có thể kết nối mạng nâng cao để truyền dữ liệu ứng dụng qua mạng, độc lập với giao thức mạng đang sử dụng. Với

Winsock, lập trình viên được cung cấp khả năng kết nối mạng nâng cao của hệ điều hành Microsoft Windows chẳng hạn như multicast (phát đa hướng) và Quality of Service (QoS – chất lượng của dịch vụ).

- Winsock theo mô hình Windows Open System Architecture (kiến trúc hệ thống mở Windows), nó xác định giao diện nhà cung cấp dịch vụ tiêu chuẩn (SPI) giữa giao diện lập trình ứng dụng (API) với các chức năng được xuất của nó và các ngăn xếp giao thức. Winsock sử dụng mô hình socket được phổ biến lần đầu tiên bởi Berkeley Software Distribution (BSD) UNIX. Sau đó, nó đã được điều chỉnh cho Windows trong Windows Sockets 1.1, trong đó các ứng dụng Windows Sockets 2 tương thích ngược. Lập trình Winsock trước đây tập trung vào TCP / IP. Một số thực tiễn lập trình hoạt động với TCP / IP không hoạt động với mọi giao thức. Do đó, API Windows Sockets 2 thêm các chức năng khi cần thiết để xử lý một số giao thức.
- Đối tượng phát triển
  - Windows Sockets 2 được thiết kế để sử dụng bởi các nhà lập trình viên C/C++. Làm quen với mạng Windows là bắt buộc.
- Các yêu cầu về môi trường run-time
  - Windows Sockets 2 có thể sử dụng trên tất cả các nền tảng Windows.

### 3. Sử dụng IP Helper

- Dưới đây là các bước để tạo viết một ứng dụng sử dụng IP Helper
  - Tạo mới một project C++ và thêm mới một source file C++ vào project.
  - Đảm bảo rằng Build environment đã đề cập đến các thư mục Include, Lib và Src của Platform Software Development Kit (SDK).
  - Đảm bảo rằng Build environment đã liên kết đến file IP Helper Library Iphlpapi.lib và Winsock Library WS2\_32.lib.
  - Bắt đầu lập trình ứng dụng IP Helper. Sử dụng IP Helper API bằng cách include các IP Helper header file vào source file.



```
#ifndef WIN32_LEAN_AND_MEAN
#define WIN32_LEAN_AND_MEAN
#endif

#include <windows.h>

#include <winsock2.h>
#include <iphlpapi.h>
#include <stdio.h>

int main() {
    return 0;
}
```

- Một số lưu ý:
  - Tập tiêu đề Iphlpapi.h (header file) là bắt buộc đối với các ứng dụng sử dụng IP Helper. Iphlpapi.h sẽ tự động bao gồm các tập tiêu đề khác với các cấu trúc (structures) và các bảng liệt kê (enumerations) được sử dụng bởi IP Helper.
  - Tập tiêu đề Winsock2.h cho Windows Sockets 2 được yêu cầu bởi hầu hết các ứng dụng sử dụng IP Helper. Dòng `#include <winsock2.h>` phải được đặt trước dòng `#include <iphlpapi.h>`
  - Tập tiêu đề winsock2.h bên trong bao gồm các thành phần chính từ tập windows.h, do đó thường không có dòng `#include <windows.h>` trong chương trình sử dụng IP Helper. Nếu cần thiết có thể dùng dòng lệnh trên nhưng phải bắt đầu bằng `#define WIN32_LEAN_AND_MEAN` macro. Vì các lý do lịch sử, windows.h mặc định đã bao gồm winsock.h cho Windows Sockets 1.1. Các khai báo trong winsock2.h sẽ xung đột với các khai báo trong windows.h. Macro WIN32\_LEAN\_AND\_MEAN sẽ ngăn chặn việc tập winsock.h được bao gồm bởi tập windows.h.

#### 4. Lệnh netstat

Lệnh netstat có chức năng hiển thị các kết nối TCP đang hoạt động, các cổng mà tại đó máy tính đang nghe và đồng thời hiển thị các thông kê Ethernet, bảng định tuyến IP, thông kê IPv4 (cho giao thức IP, ICMP, TCP và UDP) và số liệu thông kê IPv6 (cho IPv6, ICMPv6, TCP qua IPv6 và UDP qua giao thức IPv6). Nếu không có tham số, netstat sẽ hiển thị các kết nối TCP đang hoạt động.

- Cú pháp

netstat [-a] [-e] [-n] [-o] [-p <Protocol>] [-r] [-s] [<Interval>]

- Tham chiếu

Tham biến	Mô tả
-a	Hiển thị tất cả các kết nối TCP đang hoạt động, cũng như các cổng TCP và UDP mà trên đó máy tính đang nghe.
-e	Hiển thị số liệu thống kê Ethernet, chẳng hạn như số byte và các gói được gửi và nhận. Tham số này có thể được kết hợp với -s.
-n	Hiển thị các kết nối TCP đang hoạt động, tuy nhiên, các địa chỉ và số cổng được biểu thị dưới dạng số chứ không thể xác định tên cụ thể.
-o	Hiển thị các kết nối TCP đang hoạt động và bao gồm cả ID tiến trình (Process ID - PID) cho mỗi kết nối. Bạn có thể tìm ứng dụng bằng cách tra cứu PID trên tab Processes trong Windows Task Manager. Tham số này có thể được kết hợp với -a, -n và -p.
-p	Hiển thị kết nối cho giao thức được chỉ định bởi <i>Protocol</i> . Trong trường hợp này, <i>Protocol</i> có thể là tcp, udp, tcpv6 hoặc udpv6. Nếu tham số này được sử dụng với -s để hiển thị số liệu thống kê theo giao thức, <i>Protocol</i> có thể là tcp, udp, icmp, ip, tcpv6, udpv6, icmpv6 hoặc ipv6.
-s	Hiển thị số liệu thống kê theo giao thức. Theo mặc định, các số liệu thống kê được hiển thị cho các giao thức TCP, UDP, ICMP và IP. Nếu giao thức IPv6 được cài đặt, các thống kê sẽ được hiển thị cho giao thức TCP thông qua IPv6, UDP qua IPv6, ICMPv6 và IPv6. Tham số -p có thể được sử dụng để chỉ định một tập các giao thức.
-r	Hiển thị nội dung của bảng định tuyến IP (IP routing table). Thông tin này tương đương với lệnh <code>ip route</code> . Sau mỗi giây, thông tin được chọn sẽ được hiển thị lại. Nhấn CTRL + C để dừng quá trình hiển thị lại. Nếu tham số này bị bỏ qua, netstat chỉ in thông tin đã chọn một lần.
/?	Hiển thị trợ giúp tại command prompt.

- Chú giải

- Các tham số được sử dụng với lệnh này phải được bắt đầu bằng dấu gạch nối (-) thay vì dấu gạch chéo (/).
- Lệnh netstat cung cấp số liệu thống kê cho những đối tượng sau đây:

- Tên của giao thức (TCP hoặc UDP).
- Địa chỉ cục bộ. Địa chỉ IP của máy tính cục bộ và số cổng đang được sử dụng. Tên của máy tính cục bộ tương ứng với địa chỉ IP và tên của cổng được hiển thị trừ khi tham số -n được chỉ định. Nếu cổng chưa được thiết lập, số cổng được hiển thị dưới dạng dấu hoa thị (\*).
- Địa chỉ từ xa. Địa chỉ IP và số cổng của máy tính từ xa được kết nối. Các tên máy tính từ xa tương ứng với địa chỉ IP và cổng được hiển thị trừ khi tham số -n được chỉ định. Nếu cổng chưa được thiết lập, số cổng được hiển thị dưới dạng dấu hoa thị (\*).
- Trạng thái. Cho biết trạng thái của các kết nối TCP. Các trạng thái có thể như sau: CLOSE\_WAIT CLOSED ESTABLISHED FIN\_WAIT\_1 FIN\_WAIT\_2 LAST\_ACK listen SYN\_RECEIVED SYN\_SEND timeD\_WAIT để biết thêm thông tin về trạng thái của kết nối TCP, tham khảo Rfc 793.
  - Lệnh này chỉ có thể sử dụng nếu giao thức Internet Protocol (TCP/IP) được cài đặt như một thành phần trong thuộc tính của bộ điều hợp mạng trong hệ thống kết nối Mạng (Network Connections).

## **CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG**

## **CHƯƠNG 3: TRIỂN KHAI VÀ ĐÁNH GIÁ KẾT QUẢ**

### **1. Cài đặt chương trình**

### **2. Kết quả chạy chương trình**

### **3. Đánh giá kết quả**

## **KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**