**31. Implementing ARP protocols in java/C.**

**#include <stdio.h>**

**#include <stdlib.h>**

**#include <winsock2.h>**

**#include <windows.h> // Required before iphlpapi.h**

**#include <iphlpapi.h>**

**#pragma comment(lib, "ws2\_32.lib") // Link Winsock library**

**#pragma comment(lib, "iphlpapi.lib") // Link IP Helper API**

**void getMACAddress(const char \*ipAddress) {**

**DWORD dwRetVal;**

**ULONG MacAddr[2]; // Buffer for the MAC address**

**ULONG PhysAddrLen = 6; // MAC address length**

**memset(MacAddr, 0, sizeof(MacAddr)); // Initialize buffer**

**struct in\_addr DestIP;**

**DestIP.s\_addr = inet\_addr(ipAddress);**

**dwRetVal = SendARP(DestIP.s\_addr, 0, MacAddr, &PhysAddrLen);**

**if (dwRetVal == NO\_ERROR && PhysAddrLen >= 6) {**

**unsigned char \*mac = (unsigned char \*)MacAddr;**

**printf("MAC Address of %s: %02X:%02X:%02X:%02X:%02X:%02X\n",**

**ipAddress, mac[0], mac[1], mac[2], mac[3], mac[4], mac[5]);**

**} else {**

**printf("Failed to get MAC Address for %s. Ensure the IP is on the same network.\n", ipAddress);**

**}**

**}**

**int main() {**

**WSADATA wsaData;**

**if (WSAStartup(MAKEWORD(2, 2), &wsaData) != 0) {**

**printf("WSAStartup failed.\n");**

**return 1;**

**}**

**char targetIP[16];**

**printf("Enter target IP address: ");**

**scanf("%15s", targetIP); // Ensures input safety**

**getMACAddress(targetIP);**

**WSACleanup();**

**return 0;**

**}**

**A screenshot of a computer

AI-generated content may be incorrect.**