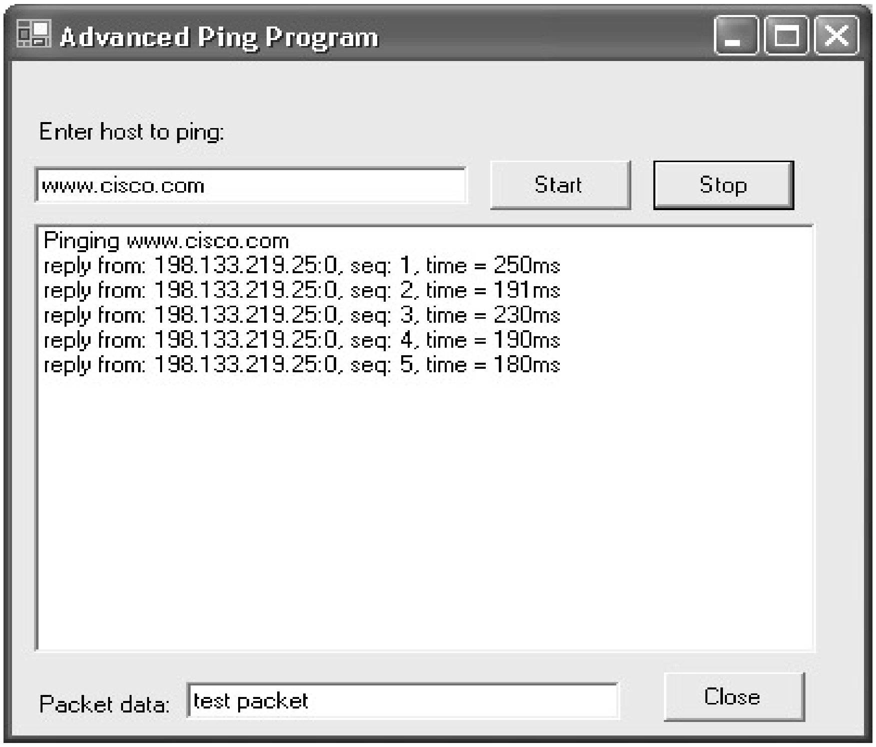
```
using System;
using System.Net;
using System.Net.Sockets;
using System.Text;
using System.Threading;

public class ICMP
{
    public byte Type;
    public byte Code;
    public UInt16 Checksum;
    public int MessageSize;
    public byte[] Message = new byte[1024];
    public ICMP()
    {
    }
    public ICMP(byte[] data, int size)
    {
        Type = data[20];
        Code = data[21];
        Checksum = BitConverter.ToUInt16(data, 22);
        MessageSize = size - 24;
        Buffer.BlockCopy(data, 24, Message, 0, MessageSize);
    }
    public byte[] getBytes()
    {
        byte[] data = new byte[MessageSize + 9];
        Buffer.BlockCopy(BitConverter.GetBytes(Type), 0, data, 0, 1);
        Buffer.BlockCopy(BitConverter.GetBytes(Code), 0, data, 1, 1);
        Buffer.BlockCopy(BitConverter.GetBytes(Checksum), 0, data, 2, 2);
        Buffer.BlockCopy(Message, 0, data, 4, MessageSize);
        return data;
    }
    public UInt16 getChecksum()
    {
        UInt32 chcksm = 0;
        byte[] data = getBytes();
        int packetsize = MessageSize + 8;
        int index = 0;
        while (index < packetsize)
        {
            chcksm += Convert.ToUInt32(BitConverter.ToUInt16(data, index));
            index += 2;
        }
        chcksm = (chcksm >> 16) + (chcksm & 0xffff);
        chcksm += (chcksm >> 16);
        return (UInt16)(~chcksm);
    }
}


namespace AdvPing_
{
    class Program
    {
        static void Main(string[] args)
        {
            int pingStart, pingStop, escTime;
            string ip = "www.vnexpress.net";
            Socket sock = new Socket(AddressFamily.InterNetwork,SocketType.Raw,
ProtocolType.Icmp);
            sock.SetSocketOption(SocketOptionLevel.Socket,
            SocketOptionName.ReceiveTimeout, 3000);
            IPHostEntry iphe = Dns.Resolve(ip);
            IPEndPoint iep = new IPEndPoint(iphe.AddressList[0], 0);
            EndPoint ep = (EndPoint)iep;
            ICMP packet = new ICMP();
            int recv, i = 1;
            packet.Type = 0x08;
            packet.Code = 0x00;
            Buffer.BlockCopy(BitConverter.GetBytes(1), 0, packet.Message, 0, 2);
            byte[] data = Encoding.ASCII.GetBytes("test packet");
            Buffer.BlockCopy(data, 0, packet.Message, 4, data.Length);
            packet.MessageSize = data.Length + 4;
            int packetsize = packet.MessageSize + 4;
            Console.WriteLine("Pinging " + ip);
            while (i <= 5)
            {
                packet.Checksum = 0;
                Buffer.BlockCopy(BitConverter.GetBytes(i), 0, packet.Message, 2, 2);
                UInt16 chcksum = packet.getChecksum();
                packet.Checksum = chcksum;
                pingStart = Environment.TickCount;
                sock.SendTo(packet.getBytes(), packetsize, SocketFlags.None, iep);
                try
                {
                    data = new byte[1024];
                    recv = sock.ReceiveFrom(data, ref ep);
                    pingStop = Environment.TickCount;
                    escTime = pingStop - pingStart;
                    Console.WriteLine("reply from: " + ep.ToString() + ", seq:" + i + ",
time= " + escTime + "ms");
                } catch (SocketException)
                {
                    Console.WriteLine("no reply from host");
                }
                i++;
                Thread.Sleep(3000);
                }
            Console.WriteLine("Ping Stoped");
            Console.ReadKey();
            }

        }
    }
```

Sinh viên tự làm thêm chương trình Advance Ping sử dụng Windows form



Enter host to ping: www.cisco.com

```
Pinging www.cisco.com
reply from: 198.133.219.25:0, seq: 1, time = 250ms
reply from: 198.133.219.25:0, seq: 2, time = 191ms
reply from: 198.133.219.25:0, seq: 3, time = 230ms
reply from: 198.133.219.25:0, seq: 4, time = 190ms
reply from: 198.133.219.25:0, seq: 5, time = 180ms
```

Packet data: test packet