CHƯƠNG 4: BÀI TẬP 4

```
using System;
using System.Net;
using System.Text;
class BinaryNetworkByteOrder
{
  public static void Main()
  {
   short test1 = 45;
   int test2 = 314159;
   long test3 = -123456789033452;
   byte[] data = new byte[1024];
   string output;
   data = BitConverter.GetBytes(test1);
   output = BitConverter.ToString(data);
   Console.WriteLine("test1 = {0}, string = {1}", test1, output);
   data = BitConverter.GetBytes(test2);
   output = BitConverter.ToString(data);
   Console.WriteLine("test2 = {0}, string = {1}", test2, output);
   data = BitConverter.GetBytes(test3);
   output = BitConverter.ToString(data);
   Console.WriteLine("test3 = {0}, string = {1}", test3, output);
   short test1b = IPAddress.HostToNetworkOrder(test1);
   data = BitConverter.GetBytes(test1b);
   output = BitConverter.ToString(data);
   Console.WriteLine("test1 = {0}, nbo = {1}", test1b, output);
   int test2b = IPAddress.HostToNetworkOrder(test2);
   data = BitConverter.GetBytes(test2b);
   output = BitConverter.ToString(data);
   Console.WriteLine("test2 = {0}, nbo = {1}", test2b, output);
   long test3b = IPAddress.HostToNetworkOrder(test3);
   data = BitConverter.GetBytes(test3b);
   output = BitConverter.ToString(data);
   Console.WriteLine("test3 = {0}, nbo = {1}", test3b, output);
  }
}
```

```
using System;
using System.Net;
using System.Net.Sockets;
using System.Text;
class NetworkOrderClient
{
  public static void Main()
  {
   byte[] data = new byte[1024];
   string stringData;
   TcpClient server;
   try
   {
     server = new TcpClient("127.0.0.1", 9050);
```

```
    } catch (SocketException)
    {
      Console.WriteLine("Unable to connect to server");
      return;
    }
    NetworkStream ns = server.GetStream();
    int recv = ns.Read(data, 0, data.Length);
    stringData = Encoding.ASCII.GetString(data, 0, recv);
    Console.WriteLine(stringData);
    short test1 = 45;
    int test2 = 314159;
    long test3 = -123456789033452;
    short test1b = IPAddress.HostToNetworkOrder(test1);
    data = BitConverter.GetBytes(test1b);
    Console.WriteLine("sending test1 = {0}", test1);
    ns.Write(data, 0, data.Length);
    ns.Flush();
    int test2b = IPAddress.HostToNetworkOrder(test2);
    data = BitConverter.GetBytes(test2b);
    Console.WriteLine("sending test2 = {0}", test2);
    ns.Write(data, 0, data.Length);
    ns.Flush();
    long test3b = IPAddress.HostToNetworkOrder(test3);
    data = BitConverter.GetBytes(test3b);
    Console.WriteLine("sending test3 = {0}", test3);
    ns.Write(data, 0, data.Length);
    ns.Flush();
    ns.Close();
    server.Close();
  }
}
```

```
using System;
using System.Net;
using System.Net.Sockets;
using System.Text;
class NetworkOrderSrvr
{
  public static void Main()
  {
    int recv;
    byte[] data = new byte[1024];
    TcpListener server = new TcpListener(9050);
    server.Start();
    Console.WriteLine("waiting for a client...");
    TcpClient client = server.AcceptTcpClient();
    NetworkStream ns = client.GetStream();
    string welcome = "Welcome to my test server";
    data = Encoding.ASCII.GetBytes(welcome);
    ns.Write(data, 0, data.Length);
    ns.Flush();
    data = new byte[2];
    recv = ns.Read(data, 0, data.Length);
```

```
    short test1t = BitConverter.ToInt16(data, 0);
    short test1 = IPAddress.NetworkToHostOrder(test1t);
    Console.WriteLine("received test1 = {0}", test1);
    data = new byte[4];
    recv = ns.Read(data, 0, data.Length);
    int test2t = BitConverter.ToInt32(data, 0);
    int test2 = IPAddress.NetworkToHostOrder(test2t);
    Console.WriteLine("received test2 = {0}", test2);
    data = new byte[8];
    recv = ns.Read(data, 0, data.Length);
    long test3t = BitConverter.ToInt64(data, 0);
    long test3 = IPAddress.NetworkToHostOrder(test3t);
    Console.WriteLine("received test3 = {0}", test3);
    ns.Close();
    client.Close();
    server.Stop();
  }
}
```