CHƯƠNG 3: BÀI TẬP 7

```csharp
using System;
using System.Net;
using System.Net.Sockets;
using System.Text;
class BestUdpClient
{
  private byte[] data = new byte[1024];
  private static IPEndPoint sender = new IPEndPoint(IPAddress.Any, 0);
  private static EndPoint Remote = (EndPoint)sender;
  private static int size = 30;
  private static int AdvSndRcvData(Socket s, byte[] message,
          EndPoint rmtdevice)
  {
   int recv = 0;
   int retry = 0;
   while (true)
   {
     Console.WriteLine("Attempt #{0}", retry);
     try
     {
      s.SendTo(message, message.Length, SocketFlags.None, rmtdevice);
      data = new byte[size];
      recv = s.ReceiveFrom(data, ref Remote);
     } catch (SocketException e)
     {
      if (e.ErrorCode == 10054)
        recv = 0;
      else if (e.ErrorCode == 10040)
      {
        Console.WriteLine("Error receiving packet");
        size += 10;
        recv = 0;
      }
     }
     if (recv > 0)
     {
      return recv;
     } else
     {
      retry++;
      if (retry > 4)
      {
        return 0;
      }
     }
   }
  }
  public static void Main()
  {
   string input, stringData;
   int recv;
   IPEndPoint ipep = new IPEndPoint(
           IPAddress.Parse("127.0.0.1"), 9050);
```

```csharp
      Socket server = new Socket(AddressFamily.InterNetwork,
               SocketType.Dgram, ProtocolType.Udp);
      int sockopt = (int)server.GetSocketOption(SocketOptionLevel.Socket,
        SocketOptionName.ReceiveTimeout);
      Console.WriteLine("Default timeout: {0}", sockopt);
      server.SetSocketOption(SocketOptionLevel.Socket,
        SocketOptionName.ReceiveTimeout, 3000);
      sockopt = (int)server.GetSocketOption(SocketOptionLevel.Socket,
        SocketOptionName.ReceiveTimeout);
      Console.WriteLine("New timeout: {0}", sockopt);
      string welcome = "Hello, are you there?";
      data = Encoding.ASCII.GetBytes(welcome);
      recv = AdvSndRcvData(server, data, ipep);
      if (recv > 0)
      {
        stringData = Encoding.ASCII.GetString(data, 0, recv);
        Console.WriteLine(stringData);
      } else
      {
        Console.WriteLine("Unable to communicate with remote host");
        return;
      }
      while(true)
      {
        input = Console.ReadLine();
        if (input == "exit")
         break;
        recv = AdvSndRcvData(server, Encoding.ASCII.GetBytes(input), ipep);
        if (recv > 0)
        {
         stringData = Encoding.ASCII.GetString(data, 0, recv);
         Console.WriteLine(stringData);
        } else
         Console.WriteLine("Did not receive an answer");
      }
      Console.WriteLine("Stopping client");
      server.Close();
    }
}
```