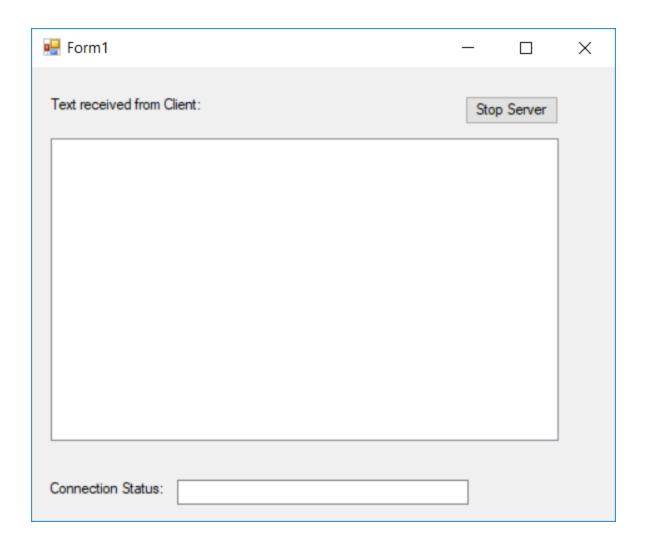
```
// Bài tập 1 – Chương 5
using System;
using System.Net;
using System.Net.Sockets;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System. Windows. Forms;
namespace AcSyncTcpSrv
  public partial class Form1 : Form
  {
    private byte[] data = new byte[1024];
    private int size = 1024;
    private Socket server;
    public Form1()
       InitializeComponent();
       server = new Socket(AddressFamily.InterNetwork,
      SocketType.Stream, ProtocolType.Tcp);
       IPEndPoint iep = new IPEndPoint(IPAddress.Any, 9050);
```

```
server.Bind(iep);
  server.Listen(5);
  server.BeginAccept(new AsyncCallback(AcceptConn), server);
}
private void btnStop_Click(object sender, EventArgs e)
  Close();
}
void AcceptConn(IAsyncResult iar)
{
  Socket oldserver = (Socket)iar.AsyncState;
  Socket client = oldserver.EndAccept(iar);
  conStatus.Text = "Connected to: " + client.RemoteEndPoint.ToString();
  string stringData = "Welcome to my server";
  byte[] message1 = Encoding.ASCII.GetBytes(stringData);
  client.BeginSend(message1, 0, message1.Length, SocketFlags.None,
      new AsyncCallback(SendData), client);
void SendData(IAsyncResult iar)
{
  Socket client = (Socket)iar.AsyncState;
  int sent = client.EndSend(iar);
  client.BeginReceive(data, 0, size, SocketFlags.None,
      new AsyncCallback(ReceiveData), client);
```

```
}
void ReceiveData(IAsyncResult iar)
{
  Socket client = (Socket)iar.AsyncState;
  int recv = client.EndReceive(iar);
  if (recv == 0)
    client.Close();
    conStatus.Text = "Waiting for client...";
    server.BeginAccept(new AsyncCallback(AcceptConn), server);
    return;
  string receivedData = Encoding.ASCII.GetString(data, 0, recv);
  results.Items.Add(receivedData);
  byte[] message2 = Encoding.ASCII.GetBytes(receivedData);
  client.BeginSend(message2, 0, message2.Length, SocketFlags.None,
      new AsyncCallback(SendData), client);
}
```



Client:

using System;

using System.Collections.Generic;

 $using\ System. Component Model;$

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System. Windows. Forms;

```
using System.Net.Sockets;
using System.Net;
namespace AcsyncTcpClients
{
  public partial class Form1 : Form
    private Socket client;
    private byte[] data = new byte[1024];
    private int size = 1024;
    public Form1()
    {
       InitializeComponent();
    }
    private void btnConnect_Click(object sender, EventArgs e)
    {
       conStatus.Text = "Connecting...";
       Socket newsock = new Socket(AddressFamily.InterNetwork,
             SocketType.Stream, ProtocolType.Tcp);
       IPEndPoint iep = new IPEndPoint(IPAddress.Parse("127.0.0.1"), 9050);
       newsock.BeginConnect(iep, new AsyncCallback(Connected), newsock);
    }
    private void btnSend_Click(object sender, EventArgs e)
```

```
{
  byte[] message = Encoding.ASCII.GetBytes(newText.Text);
  newText.Clear();
  client.BeginSend(message, 0, message.Length, SocketFlags.None,
      new AsyncCallback(SendData), client);
}
private void btnDisconnect_Click(object sender, EventArgs e)
{
  client.Close();
  conStatus.Text = "Disconnected";
}
void Connected(IAsyncResult iar)
{
  client = (Socket)iar.AsyncState;
  try
    client.EndConnect(iar);
    conStatus.Text = "Connected to: " + client.RemoteEndPoint.ToString();
    client.BeginReceive(data, 0, size, SocketFlags.None,
         new AsyncCallback(ReceiveData), client);
  catch (SocketException)
```

```
MessageBox.Show("Không thể kết nối đến Server");
    conStatus.Text = "Error connecting";
  }
void ReceiveData(IAsyncResult iar)
  Socket remote = (Socket)iar.AsyncState;
  int recv = remote.EndReceive(iar);
  string stringData = Encoding.ASCII.GetString(data, 0, recv);
  results.Items.Add(stringData);
void SendData(IAsyncResult iar)
  Socket remote = (Socket)iar.AsyncState;
  int sent = remote.EndSend(iar);
  remote.BeginReceive(data, 0, size, SocketFlags.None,
      new AsyncCallback(ReceiveData), remote);
}
```

