**Name: Rana Hamza Ali**

**St.Id : 9834**

**Assignment 1**

**Server**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Net;

using System.Net.Sockets;

using System.IO;

namespace assignment1Server

{

    class Program

    {

        public static void data(int x1, int y1, int x2, int y2, int x3, int y3)

        {

            UdpClient ip1 = new UdpClient("192.168.4.3", 1200);

            UdpClient ip2 = new UdpClient("192.168.4.3", 1300);

            byte[] data1 = Encoding.Unicode.GetBytes(x1 + "," + y1 + "," + x2 + "," + y2 + "," + x3 + "," + y3);

            ip1.Send(data1, data1.Length);

            byte[] data2 = Encoding.Unicode.GetBytes(x2 + "," + y2 + "," + x1 + "," + y1 + "," + x3 + "," + y3);

            ip2.Send(data2, data2.Length);

            Console.Clear();

            Console.SetCursorPosition(x1, y1);

            Console.WriteLine("C2");

            Console.SetCursorPosition(x2, y2);

            Console.WriteLine("C1");

            Console.SetCursorPosition(x3, y3);

            Console.WriteLine("C");

        }

        static void Main(string[] args)

        {

            int x1 = 3;

            int y1 = 3;

            int x2 = 4;

            int y2 = 5;

            int x3 = 1;

            int y3 = 1;

            try

            {

                Program.data(x1, y1, x2, y2, x3, y3);

                while (true)

                {

                    var keyRead = Console.ReadKey(true);

                    try

                    {

                        switch (keyRead.Key)

                        {

                            case ConsoleKey.UpArrow:

                                y3--;

                                Program.data(x1, y1, x2, y2, x3, y3);

                                break;

                            case ConsoleKey.DownArrow:

                                y3++;

                                Program.data(x1, y1, x2, y2, x3, y3);

                                break;

                            case ConsoleKey.RightArrow:

                                x3++;

                                Program.data(x1, y1, x2, y2, x3, y3);

                                break;

                            case ConsoleKey.LeftArrow:

                                x3--;

                                Program.data(x1, y1, x2, y2, x3, y3);

                                break;

                        }

                    }

                    catch

                    {

                        y3 = 0;

                        x3 = 1;

                    }

                }

            }

            catch (Exception e)

            {

                Console.WriteLine(e);

            }

        }

    }

}

**Client**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Net;

using System.Net.Sockets;

using System.Threading;

using System.IO;

namespace assignment1Client

{

    class Program

    {

        public static int x1;

        public static int y1;

        public static int x2;

        public static int y2;

        public static int x3;

        public static int y3;

        public static void recivedata(IPEndPoint ip, UdpClient server)

        {

            byte[] data = server.Receive(ref ip);

            string datarecieved = Encoding.Unicode.GetString(data, 0, data.Length);

            string[] slit\_data = datarecieved.Split(',');

            int[] ch = new int[6];

            for (int i = 0; i < 6; i++)

            {

                ch[i] = int.Parse(slit\_data[i]);

            }

            Console.Clear();

            Console.SetCursorPosition(ch[0], ch[1]);

            x2 = ch[0];

            y2 = ch[1];

            Console.WriteLine("C1");

            Console.SetCursorPosition(ch[2], ch[3]);

            x1 = ch[2];

            y1 = ch[3];

            Console.WriteLine("C2");

            Console.SetCursorPosition(ch[4], ch[5]);

            x3 = ch[4];

            y3 = ch[5];

            Console.WriteLine("C");

        }

        static void Main(string[] args)

        {

            try

            {

                IPEndPoint ip = new IPEndPoint(IPAddress.Any, 1200);

                UdpClient server = new UdpClient(ip);

                while (true)

                {

                    Program.recivedata(ip, server);

                }

            }

            catch

            {

                Console.WriteLine("Connection failed");

                Console.ReadKey();

            }

        }

    }

}