Assignment 1: Simplest Client-Server

By Trung Le – 101264698

* GitHub links:
  + <https://github.com/KojimaMcMaple/GAME-3110_SimplestServer>
  + <https://github.com/KojimaMcMaple/GAME-3110_SimplestClient>
* What is the “out” keyword parameter modifier in C#? What does it do?
* The “**out**” keyword allows the arguments passed in a function to be passed by reference, meaning you can change the value of the arguments after running the function by using “**out**”.
* Why do we need to use the out keyword parameter mod with int values but not with references to class instances?
* If we pass the reference to a class instance, the function can change what the reference refers to, like making it point to a different object, which is not what we want.

1. What are the categories that the QosType enumeration describes?

* In the codes, we’re using “QosType.Reliable” and “QosType.Unreliable” to configure the 2 channels needed for the connection:
  + QosType.Reliable: Messages are guaranteed to be delivered, but they don’t have to be in order.
  + QosType.Unreliable: No guarantee of delivery or ordering.

1. What does the NetworkTransport.Init function do?

* It initializes the NetworkTransport object, which is required for the next steps of networking.

1. What does the NetworkTransport.Disconnect function do?

* It aims to close the connection by sending a disconnect signal to the connected peer/computer. If the signal fails to send, connection is just closed by timeout.

1. What does the Encoding.Unicode.GetString and the Encoding.Unicode.GetBytes functions do?

* They encode the variable passed into either string (GetString) or bytes (GetBytes) using the Unicode standard, because NetworkTransport.Send/Receive uses bytes.

1. What does the sizeof operator do?

* It returns the size in bytes. In the context of “msg.Length \* sizeof(char)”, we get the byte size of 1 char and multiply it with the number of chars in msg (msg.Length) to find the byte size of msg.

1. What does the NetworkEventType enumeration describe?

* It describes the event that is returned from NetworkTransport.Receive, we use it to capture and process the different network events.

1. What does the NetworkTransport.Receive function do? What does each param do?

* It’s a call to poll network events. The params are:
  + hostId: Host ID associated with the event.
  + connectionId: The connectionID that received the event.
  + channelId: The channel ID associated with the event.
  + buffer: The buffer that will hold the data received.
  + bufferSize: Size of the buffer supplied.
  + receivedSize: The actual receive size of the data.
  + error: Error (can be cast to NetworkError for more information).

1. What does the NetworkTransport.Send function do? What does each param do?

* It’s a call to send data to the connected computer. The params are:
  + hostId: Host ID associated with this connection.
  + connectionId: ID of the connection.
  + channelId: The channel ID to send on.
  + buffer: Buffer containing the data to send.
  + size: Size of the buffer.
  + error: Error (can be cast to NetworkError for more information).

1. What is the ConnectionConfig class? What does the AddChannel function of the ConnectionConfig class do?

* It defines the parameters for the connection between 2 machines. AddChannel adds a new channel to ConnectionConfig, and returns the id of said channel.

1. What is the HostTopology class? What does the AddHost function of the HostTopology class do?

* It defines the network topology of the host. AddHost creates a host based on the HostTopology.

1. Read through the code, search for any internal class/function/enumeration/ect that you do not understand and then ask the question, what does it do.

* They’re all clear to me, but seeing “Method group is Obsolete” on the documentation is concerning.