**SOLVING THE UNIT TEST ERRORS**

**1) SerializeCellTest (1111,22221,11111,2221)**

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Error message:** Test method UnitTestsProject.HTMSerializationTests.SerializeCellTest threw exception: System.FormatException: Input string was not in a correct format.

\***HTMSerializationTests.cs (line 552)**

public void SerializeCellTest(int parentIndx, int colSeq, int cellsPerCol, int cellId)

{

Cell cell = new Cell(12, 14, 16, new CellActivity());

var distSeg1 = new DistalDendrite(cell, 1, 2, 2, 1.0, 100);

cell.DistalDendrites.Add(distSeg1);

var distSeg2 = new DistalDendrite(cell, 44, 24, 34, 1.0, 100);

cell.DistalDendrites.Add(distSeg2);

Cell preSynapticcell = new Cell(11, 14, 16, new CellActivity());

var synapse1 = new Synapse(cell, distSeg1.SegmentIndex, 23, 1.0);

preSynapticcell.ReceptorSynapses.Add(synapse1);

var synapse2 = new Synapse(cell, distSeg2.SegmentIndex, 27, 1.0);

preSynapticcell.ReceptorSynapses.Add(synapse2);

using (StreamWriter sw = new StreamWriter($"ser\_{nameof(SerializeCellTest)}.txt"))

{

cell.SerializeT(sw);

}

using (StreamReader sr = new StreamReader($"ser\_{nameof(SerializeCellTest)}.txt"))

{

HtmSerializer ser = new HtmSerializer();

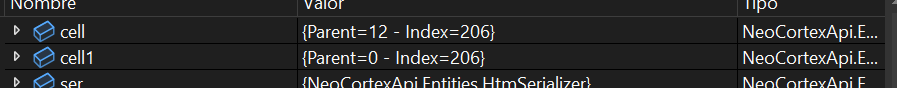
Cell cell1 = ser.DeserializeCell(sr);

Assert.IsTrue(cell1.Equals(cell));

}

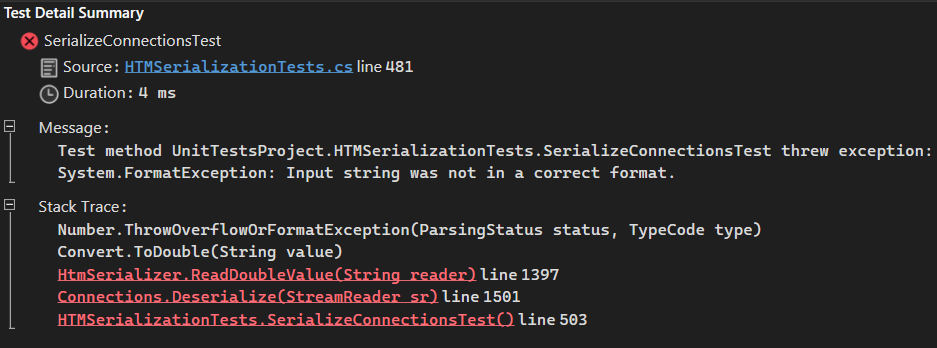
}

**Way to solve 1: the parameter parent for our cell is not the same as cell1 so there is being an error in Assert.IsTrue when comparing the two cells. So, the parameter parentIndx should be 0 not 12 as it was written.**





**2) SerializeConnectionsTest**



**Error message**: Test method UnitTestsProject.HTMSerializationTests.SerializeConnectionsTest threw exception: System.FormatException: Input string was not in a correct format.

\***HTMSerializationTests.cs (line 481)**

public void SerializeConnectionsTest()

{

int[] inputDims = { 3, 4, 5 };

int[] columnDims = { 35, 43, 52 };

HtmConfig cfg = new HtmConfig(inputDims, columnDims);

Connections connections = new Connections(cfg);

Cell cells = new Cell(12, 14, 16, new CellActivity());

var distSeg1 = new DistalDendrite(cells, 1, 2, 2, 1.0, 100);

var distSeg2 = new DistalDendrite(cells, 44, 24, 34, 1.0, 100);

connections.ActiveSegments.Add(distSeg1);

using (StreamWriter sw = new StreamWriter($"ser\_{nameof(SerializeConnectionsTest)}.txt"))

{

connections.Serialize(sw);

}

using (StreamReader sr = new StreamReader($"ser\_{nameof(SerializeConnectionsTest)}.txt"))

{

Connections connections1 = Connections.Deserialize(sr);

Assert.IsTrue(connections.Equals(connections1));

}

}

**3) SerializeDictionarystringintA**

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Error message:** Assert.IsTrue failed.

**\*HTMSerializationTests.cs (line 368)**

public void SerializeDictionarystringintA()

{

HtmSerializer htm = new HtmSerializer();

Dictionary<String, int[]> keyValues = new Dictionary<String, int[]>

{

{ "Hello", new int[] { 1, 2, 3 } },

{ "GoodMorning", new int[] { 4, 5, 6 } },

{ "Goodevening", new int[] { 7, 8, 9 } }

};

Dictionary<String, int[]> keyValuePairs = new Dictionary<String, int[]>();

using (StreamWriter sw = new StreamWriter($"ser\_{nameof(SerializeDictionarystringintA)}.txt"))

{

htm.SerializeBegin("UnitTest", sw);

htm.SerializeValue(keyValues, sw);

htm.SerializeEnd("UnitTest", sw);

}

using (StreamReader sr = new StreamReader($"ser\_{nameof(SerializeDictionarystringintA)}.txt"))

{

while (sr.Peek() >= 0)

{

string data = sr.ReadLine();

if (data == String.Empty || data == htm.ReadBegin("UnitTest"))

{

}

else if (data == htm.ReadEnd("UnitTest"))

{

break;

}

else

{

string[] str = data.Split(HtmSerializer.ParameterDelimiter);

for (int i = 0; i < str.Length; i++)

{

switch (i)

{

case 0:

keyValuePairs = htm.ReadDictSIarray(str[i]);

break;

default:

break;

}

}

}

}

}

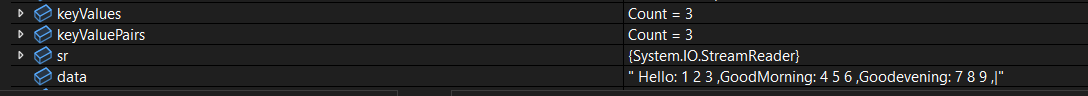
Assert.IsTrue(keyValuePairs.SequenceEqual(keyValues));

}

**Way to solve 3: We can get “**Assert.IsTrue(keyValuePairs.SequenceEqual(keyValues));**” code line into the switch case loop. Because we need to verify that the Assert.IsTrue statement at the end of the method is correctly comparing the key value pairs. Depending on the implementation of the HtmSerializer, the comparison may need to be modified to account for any differences in the serialized and deserialized data.**

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu



**4) SerializeDistalDendrite (1,2,2,1,100)**

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Error message:** Test method UnitTestsProject.HTMSerializationTests.SerializeDistalDendrite threw exception: System.NullReferenceException: Object reference not set to an instance of an object.

**\* HTMSerializationTests.cs (line 590)**

public void SerializeDistalDendrite(int flatIdx, long lastUsedIteration, int ordinal, double synapsePermConnected, int numInputs)

{

Cell cell = new Cell(12, 14, 16, new CellActivity());

var distSeg1 = new DistalDendrite(cell, 1, 2, 2, 1.0, 100);

cell.DistalDendrites.Add(distSeg1);

var distSeg2 = new DistalDendrite(cell, 44, 24, 34, 1.0, 100);

cell.DistalDendrites.Add(distSeg2);

Cell preSynapticcell = new Cell(11, 14, 16, new CellActivity());

var synapse1 = new Synapse(cell, distSeg1.SegmentIndex, 23, 1.0);

preSynapticcell.ReceptorSynapses.Add(synapse1);

var synapse2 = new Synapse(cell, distSeg2.SegmentIndex, 27, 1.0);

preSynapticcell.ReceptorSynapses.Add(synapse2);

// Serializes the segment to file.

using (StreamWriter sw = new StreamWriter($"ser\_{nameof(SerializeDistalDendrite)}.txt"))

{

distSeg1.Serialize(sw);

}

using (StreamReader sr = new StreamReader($"ser\_{nameof(SerializeDistalDendrite)}.txt"))

{

HtmSerializer ser = new HtmSerializer();

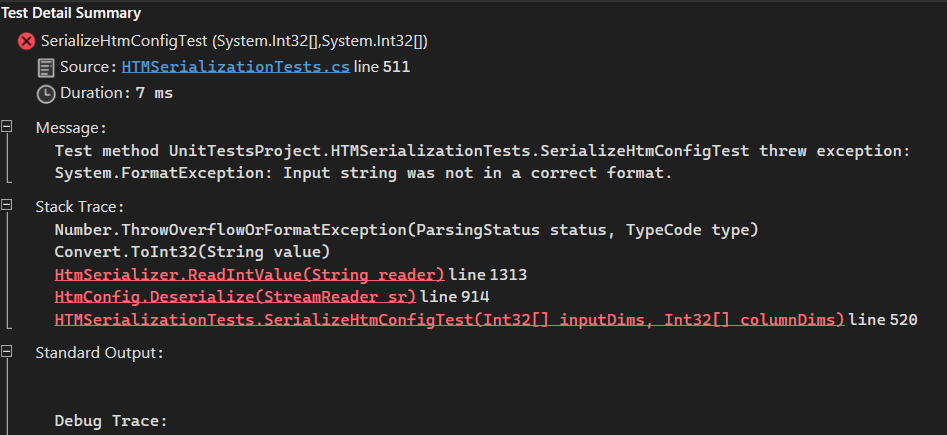
DistalDendrite distSegment1 = ser.DeserializeDistalDendrite(sr);

Assert.IsTrue(distSegment1.Equals(distSeg1));

}

}

**5) SerializeHtmConfigTest (System.Int32[], System.Int32[])**



**Error message:** Test method UnitTestsProject.HTMSerializationTests.SerializeHtmConfigTest threw exception: System.FormatException: Input string was not in a correct format.

**\* HTMSerializationTests.cs (line 511)**

public void SerializeHtmConfigTest(int[] inputDims, int[] columnDims)

{

HtmConfig matrix = new HtmConfig(inputDims, columnDims);

using (StreamWriter sw = new StreamWriter($"ser\_{nameof(SerializeHtmConfigTest)}.txt"))

{

matrix.Serialize(sw);

}

using (StreamReader sr = new StreamReader($"ser\_{nameof(SerializeHtmConfigTest)}.txt"))

{

HtmConfig matrix1 = HtmConfig.Deserialize(sr);

Assert.IsTrue(matrix.Equals(matrix1));

}

}

**6) SerializeInMemoryDistributedDictionary**

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Error message:** Test method UnitTestsProject.HTMSerializationTests.SerializeInMemoryDistributedDictionary threw exception: System.FormatException: Input string was not in a correct format.

**\* HTMSerializationTests.cs (line 879)**

public void SerializeInMemoryDistributedDictionary()

{

InMemoryDistributedDictionary<int, int> numNodes = new InMemoryDistributedDictionary<int, int>(3);

// There are no Serialize of Dictionary in InMemoryDistributedDictionary

numNodes.Add(145, 29);

numNodes.Add(123, 26);

numNodes.Add(531, 26);

numNodes.Add(1536, 26);

numNodes.Add(1529, 26);

// Serialize

using (StreamWriter sw = new StreamWriter("InMem.txt"))

{

numNodes.Serialize(sw);

}

// Deserialize

InMemoryDistributedDictionary<int, int> newTest = new InMemoryDistributedDictionary<int, int>();

using (StreamReader sr = new StreamReader("InMem.txt"))

{

newTest = InMemoryDistributedDictionary<int, int>.Deserialize(sr);

HtmSerializer.IsEqual(numNodes, newTest);

}

}

**7) SerializeProximalDendriteTest (0,12,34,23)**

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Error message:** Test method UnitTestsProject.HTMSerializationTests.SerializeProximalDendriteTest threw exception: System.FormatException: Input string was not in a correct format.

**\* HTMSerializationTests.cs (line 721)**

public void SerializeProximalDendriteTest(int colIndx, double synapsePermConnected, int numInputs)

{

ProximalDendrite proximal = new ProximalDendrite(colIndx, synapsePermConnected, numInputs);

var rfPool = new Pool(1, 28);

Cell cell = new Cell(12, 14, 16, new CellActivity());

var distSeg1 = new DistalDendrite(cell, 1, 2, 2, 1.0, 100);

cell.DistalDendrites.Add(distSeg1);

var distSeg2 = new DistalDendrite(cell, 44, 24, 34, 1.0, 100);

cell.DistalDendrites.Add(distSeg2);

Cell preSynapticcell = new Cell(11, 14, 16, new CellActivity());

var synapse1 = new Synapse(cell, distSeg1.SegmentIndex, 23, 1.0);

preSynapticcell.ReceptorSynapses.Add(synapse1);

var synapse2 = new Synapse(cell, distSeg2.SegmentIndex, 27, 1.0);

preSynapticcell.ReceptorSynapses.Add(synapse2);

rfPool.m\_SynapsesBySourceIndex = new Dictionary<int, Synapse>();

rfPool.m\_SynapsesBySourceIndex.Add(3, synapse1);

rfPool.m\_SynapsesBySourceIndex.Add(67, synapse2);

proximal.RFPool = rfPool;

//HtmSerializer2 htm = new HtmSerializer2();

using (StreamWriter sw = new StreamWriter($"ser\_{nameof(SerializeProximalDendriteTest)}.txt"))

{

proximal.Serialize(sw);

}

//htm.indent($"ser\_{nameof(SerializeProximalDendriteTest)}.txt");

using (StreamReader sr = new StreamReader($"ser\_{nameof(SerializeProximalDendriteTest)}.txt"))

{

ProximalDendrite proximal1 = ProximalDendrite.Deserialize(sr);

Assert.IsTrue(proximal.Equals(proximal1));

}

}

**8) SerializeSparseBinaryMatrix**

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Error message:** Test method UnitTestsProject.HTMSerializationTests.SerializeSparseBinaryMatrix threw exception: System.FormatException: Input string was not in a correct format.

**\* HTMSerializationTests.cs (line 904)**

public void SerializeSparseBinaryMatrix()

{

// Create SParse BinarySparseMatrix

// either by dicrect creation or running experiment

int[] dimensions = { 100, 100 };

//IDistributedDictionary<int, int[]> dict = new();

SparseBinaryMatrix binaryMatrix = new(dimensions, false);

// Serialize

using (StreamWriter sw = new StreamWriter("Binary.txt"))

{

binaryMatrix.Serialize(sw);

}

// Deserizlize

SparseBinaryMatrix newBinary = new();

using (StreamReader sr = new StreamReader("Binary.txt"))

{

newBinary = SparseBinaryMatrix.Deserialize(sr);

HtmSerializer.IsEqual(binaryMatrix, newBinary);

}

}

**9) SerializeSynapseTest (13,87,22,45)**

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Error message:** Test method UnitTestsProject.HTMSerializationTests.SerializeSynapseTest threw exception: System.FormatException: Input string was not in a correct format.

**\* HTMSerializationTests.cs (line 631)**

public void SerializeSynapseTest(int segmentindex, int synapseindex, double permanence)

{

Cell cell = new Cell(12, 14, 16, new CellActivity());

var distSeg1 = new DistalDendrite(cell, 1, 2, 2, 1.0, 100);

cell.DistalDendrites.Add(distSeg1);

var distSeg2 = new DistalDendrite(cell, 44, 24, 34, 1.0, 100);

cell.DistalDendrites.Add(distSeg2);

Cell preSynapticcell = new Cell(11, 14, 16, new CellActivity());

var synapse1 = new Synapse(cell, distSeg1.SegmentIndex, 23, 1.0);

preSynapticcell.ReceptorSynapses.Add(synapse1);

var synapse2 = new Synapse(cell, distSeg2.SegmentIndex, 27, 1.0);

preSynapticcell.ReceptorSynapses.Add(synapse2);

using (StreamWriter sw = new StreamWriter($"ser\_{nameof(SerializeSynapseTest)}.txt"))

{

synapse1.Serialize(sw);

}

using (StreamReader sr = new StreamReader($"ser\_{nameof(SerializeSynapseTest)}.txt"))

{

HtmSerializer ser = new HtmSerializer();

Synapse synapseT1 = ser.DeserializeSynapse(sr);

Assert.IsTrue(synapse1.Equals(synapseT1));

}

}

**10) Serialize Pool Test (3) – (group error)**

a) SerializePoolTest (0,34)

b) SerializePoolTest (1,28)

c) SerializePoolTest (1000,3426)

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Error message:** Test method UnitTestsProject.HTMSerializationTests.SerializePoolTest threw exception: System.FormatException: Input string was not in a correct format.

- Error is same in a,b,c.

**\* HTMSerializationTests.cs (line 674)**

public void SerializePoolTest(int size, int numInputs)

{

Pool pool = new Pool(size, numInputs);

//pool.m\_SynapseConnections = new List<int>();

//pool.m\_SynapseConnections.Add(34);

//pool.m\_SynapseConnections.Add(87);

//pool.m\_SynapseConnections.Add(44);

Cell cell = new Cell(12, 14, 16, new CellActivity());

var distSeg1 = new DistalDendrite(cell, 1, 2, 3, 1.0, 100);

cell.DistalDendrites.Add(distSeg1);

var distSeg2 = new DistalDendrite(cell, 44, 24, 4, 1.0, 100);

cell.DistalDendrites.Add(distSeg2);

Cell preSynapticcell = new Cell(11, 14, 16, new CellActivity());

var synapse1 = new Synapse(cell, distSeg1.SegmentIndex, 23, 1.0);

preSynapticcell.ReceptorSynapses.Add(synapse1);

var synapse2 = new Synapse(cell, distSeg2.SegmentIndex, 27, 1.0);

preSynapticcell.ReceptorSynapses.Add(synapse2);

pool.m\_SynapsesBySourceIndex = new Dictionary<int, Synapse>();

pool.m\_SynapsesBySourceIndex.Add(3, synapse1);

pool.m\_SynapsesBySourceIndex.Add(67, synapse2);

using (StreamWriter sw = new StreamWriter($"ser\_{nameof(SerializePoolTest)}.txt"))

{

pool.Serialize(sw);

}

using (StreamReader sr = new StreamReader($"ser\_{nameof(SerializePoolTest)}.txt"))

{

Pool pool1 = Pool.Deserialize(sr);

Assert.IsTrue(pool1.Equals(pool));

}

}

---**Final note: We need to solve these unit test errors for our code to work.**