**After this line of code:**

Queue nums = new Queue(); // assume that the array is 3 elements long

**The Queue object will look like:**

|  |  |  |  |
| --- | --- | --- | --- |
| **int [] items** | | | |
| **Index 🡪** | **0** | **1** | **2** |
| **Element at that index 🡪** | **0** | **0** | **0** |

|  |  |  |
| --- | --- | --- |
| **int count** | **int frontOfQueue** | **int backOfQueue** |
| **0** | **0** | **0** |

**Mentally trace through this code:**

if (nums.isEmpty() )

Console.WriteLine("Queue started out empty");

else

Console.WriteLine("Queue started out NON empty");

**Mentally trace through this code:**

int x;

nums.Peek(out x); // what is the return value?

**After this code:**

nums.Enqueue(10); // what is the return value?

**The Queue object will look like:**

|  |  |  |  |
| --- | --- | --- | --- |
| **int [] items** | | | |
| **Index 🡪** | **0** | **1** | **2** |
| **Element at that index 🡪** |  |  |  |

|  |  |  |
| --- | --- | --- |
| **int count** | **int frontOfQueue** | **int backOfQueue** |
|  |  |  |

**After this code:**

nums.Enqueue(20); // what is the return value?

nums.Enqueue(30); // what is the return value?

**The Queue object will look like:**

|  |  |  |  |
| --- | --- | --- | --- |
| **int [] items** | | | |
| **Index 🡪** | **0** | **1** | **2** |
| **Element at that index 🡪** |  |  |  |

|  |  |  |
| --- | --- | --- |
| **int count** | **int frontOfQueue** | **int backOfQueue** |
|  |  |  |

**Mentally trace through this code:**

nums.Enqueue(40); // what is the return value?

**Mentally trace through this code:**

if (nums.isEmpty() )

Console.WriteLine("Queue empty after 3 Enqueues!");

else

Console.WriteLine("Queue contains stuff after 3 Enqueues!");

nums.Peek(out x); // what is the return value?

Console.WriteLine("front element is {0}", x);

**After this code:**

nums.Dequeue(out x); // what is the return value?

Console.WriteLine("top most element WAS {0}, but we just removed it", x);

**The Queue object will look like:**

|  |  |  |  |
| --- | --- | --- | --- |
| **int [] items** | | | |
| **Index 🡪** | **0** | **1** | **2** |
| **Element at that index 🡪** |  |  |  |

|  |  |  |
| --- | --- | --- |
| **int count** | **int frontOfQueue** | **int backOfQueue** |
|  |  |  |

**After this code:**

nums.Dequeue(out x); // we’re going to ignore x

nums.Dequeue(out x);

Console.WriteLine("last element is {0}", x);

**The Queue object will look like:**

|  |  |  |  |
| --- | --- | --- | --- |
| **int [] items** | | | |
| **Index 🡪** | **0** | **1** | **2** |
| **Element at that index 🡪** |  |  |  |

|  |  |  |
| --- | --- | --- |
| **int count** | **int frontOfQueue** | **int backOfQueue** |
|  |  |  |

**Mentally trace through this code:**

nums.Dequeue(out x); // what is the return value? What is x?

**Mentally trace through this code:**

if (nums.isEmpty() )

Console.WriteLine("Queue empty!");

else

Console.WriteLine("Queue contains stuff!");