**Topic 2: Copy and Compare Arrays**

Top of Form

Bottom of Form

**Content**

**Reading**

[Reading](https://dmacc.blackboard.com/webapps/blackboard/content/listContent.jsp?course_id=_102593_1&content_id=_7286837_1)

Array copy

* + TutorialsPoint: tutorialspoint.com
    - [Array Copy in C#](https://www.tutorialspoint.com/Array-Copy-in-Chash)

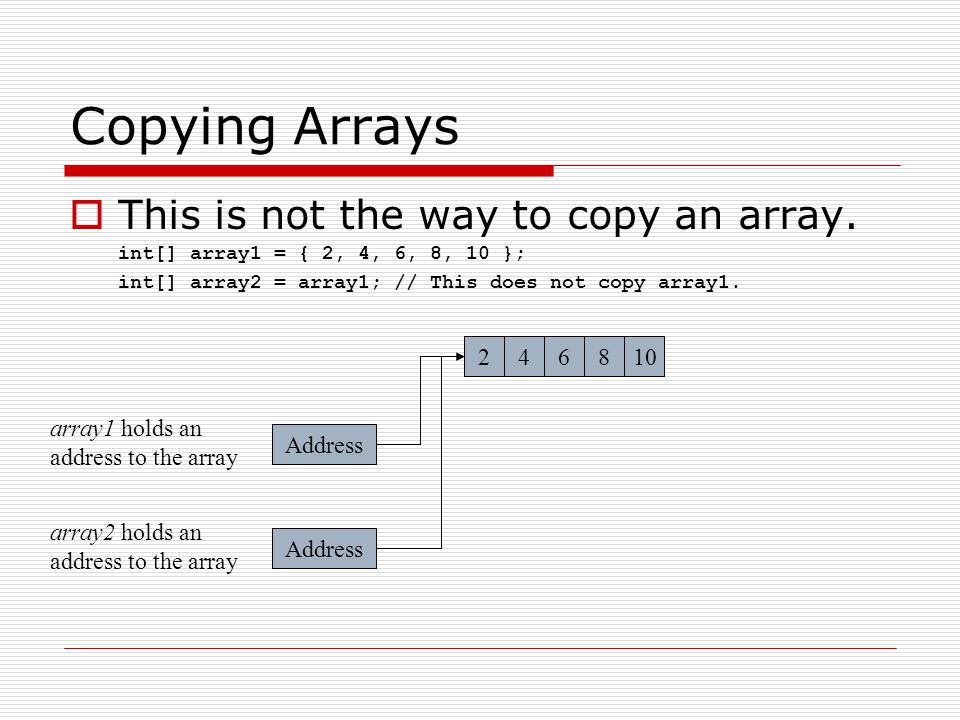
Array compare

* + TutorialsPoint: tutorialspoint.com
    - [How to compare two arrays in C#](https://www.tutorialspoint.com/How-to-compare-two-arrays-in-Chash)

**Copy Arrays**

[Copy Arrays](https://dmacc.blackboard.com/webapps/blackboard/content/listContent.jsp?course_id=_102593_1&content_id=_7286837_1)

How do we copy arrays? Remember they are objects and have reference variables. So using = does not work. You will end up with 2 reference variables to the same array.



So what can you do?

Create the second array (declare it) so it is in memory. Then use your magic loop powers to copy the elements one at a time.

int[] sourceArray = { 2, 4, 6, 8 };

int[] targetArray = new int[sourceArray.length];

for (int index = 0; index < sourceArray.length; index++){

targetArray[index] = sourceArray[index];

**Comparing arrays & foreach**

[Comparing arrays & foreach](https://dmacc.blackboard.com/webapps/blackboard/content/listContent.jsp?course_id=_102593_1&content_id=_7286837_1)

https://youtu.be/TcxHjhcBc5o

using System;

namespace Module6

{

class Program

{

static void Main(string[] args)

{

// Declare and Initialize an array

int[] age = new int[3];

int[] size = new int[3] { 2, 3, 1 };

age[0] = 2;

age[1] = 3;

age[2] = 12;

// using foreach

foreach (int item in age)

{

Console.WriteLine(item);

}

foreach (int item in size)

{

Console.WriteLine(item);

}

// comparing arrays item by item

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

{

if (age[i] == size[i])

{

Console.WriteLine("same!" + age[i]);

}

else

{

Console.WriteLine("age[" + i + "] =" + age[i] + " DOES NOT EQUAL " + "size[" + i + "]=" + size[i]);

}

}

}

}

}

**Compare Arrays**

[Compare Arrays](https://dmacc.blackboard.com/webapps/blackboard/content/listContent.jsp?course_id=_102593_1&content_id=_7286837_1)

When you compare arrays, you cannot use == as you did for numbers (and strings in this crazy C# language) or even .Equals.

You can use your loop power to write you own algorithm to loop over both to see if each element is equal. The first check would be if they are the same length. If one has 3 elements and the other has 5, they are clearly not equal.

Stackoverflow (one of your best friends) has some interesting solutions:

<https://stackoverflow.com/questions/3232744/easiest-way-to-compare-arrays-in-c-sharp>

**Arrays Compare & Copy Practice**

[Arrays Compare & Copy Practice](https://dmacc.blackboard.com/webapps/blackboard/content/listContent.jsp?course_id=_102593_1&content_id=_7286837_1)

[UnitTest1.cs](https://dmacc.blackboard.com/bbcswebdav/pid-7286866-dt-content-rid-101466159_1/xid-101466159_1)   
 Write two methods to handle arrays. Name your solution and namespace Module6, name your class CopyCompare. Don't forget to make it public for Unit Testing. Your file can be CopyCompare.cs or Program.cs.

One useful property that arrays have is Length. For an array named strArray, you can find its length with strArray.Length. Use this property in your methods.

* + Write a method CopyArray()
    - Accepts an array of integer, pass by value
    - Uses for loop to copy one array into a second array
    - Returns an array, a copy of the array passed in
  + Write a method returns true or false called CompareArrays()
    - Accepts two arrays of type string, pass by value
    - Checks for length equivalence (Use .Length property!), exits if the are not of the same length
    - Uses for to compare element by element each of the two arrays.
  + Main() will call both methods
    - Declare two integer arrays inventory and inventoryCheck
      * Initialize inventory to size 5 with random integers of your choice.
      * Set inventoryCheck to size 5 with different random integers of your choice
    - Call compareArray() to check if the arrays are equal and print the result (HINT: you might need an if statement)
    - Declare a third integer array, called copyOfInventoryCheck and copy the contents of inventoryCheck using copyArray()
    - Print all three arrays using foreach loop(s), be sure the use knows which array is being printed
  + Unit Test
    - Test with attached Unit Tests

Try it out before checking a possible solution: [Program.cs](https://dmacc.blackboard.com/bbcswebdav/pid-7286866-dt-content-rid-101466160_1/xid-101466160_1)