**Question 1: Implement a Dynamic Array Class**

**Problem Statement:**  
Create a class DynamicArray<T> that implements a dynamic array with the following methods:

* void Add(T item): Adds an element to the array, doubling the size when capacity is exceeded.
* void RemoveAt(int index): Removes an element at the specified index, shifting elements accordingly.
* T Get(int index): Returns the element at the given index.
* int Count { get; }: Returns the number of elements in the array.
* int Capacity { get; }: Returns the current capacity of the array.

Write a **main method** to demonstrate its functionality.

**Question 2: Resize a Dynamic Array**

**Problem Statement:**  
Write a function ResizeArray<T>(T[] arr, int newSize) that takes an array and resizes it to newSize. The function should:

* Allocate a new array of the specified size.
* Copy elements from the original array to the new array.
* Return the resized array.

Demonstrate the function by resizing an integer array from size 5 to size 10.

**Question 3: Implement a Stack using a Dynamic Array**

**Problem Statement:**  
Implement a **Stack** using a dynamic array in C#. The stack should have the following methods:

* void Push(T item): Adds an item to the stack (expanding capacity if needed).
* T Pop(): Removes and returns the top element of the stack.
* T Peek(): Returns the top element without removing it.
* bool IsEmpty(): Returns true if the stack is empty, otherwise false.

Write a main method to test the implementation.

**Question 4: Merge Two Dynamic Arrays**

**Problem Statement:**  
Write a C# program to merge two dynamic arrays of integers into a **new sorted array**

**Question 5: Find Missing Numbers in a Dynamic Array**

**Problem Statement:**  
Given an array of **N** integers where numbers are in the range **1 to N+M** (some numbers are missing), find all missing numbers.