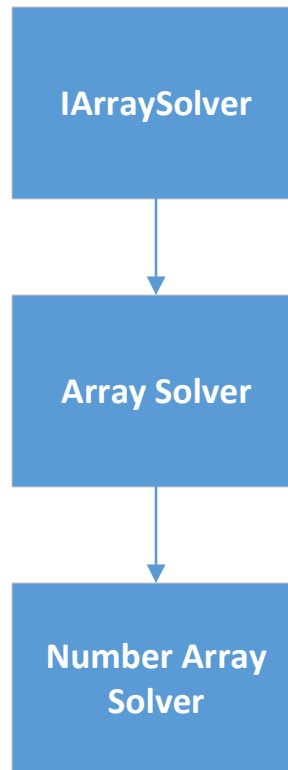# Class Diagram


IArraySolver – An interface for the array solver Contains the SolveNumberArray method and could be used to extend other methods.

ArraySolver – An abstract class used to create custom logic in the child classes.

NumberSolver – A child class of the arraySolver class. Main function is to find a missing number in the specified number array.

Start

Main Program → **End**

NumberArraySolver.SolveNumberArray()

NumberArraySolver.GetMissingNumber()
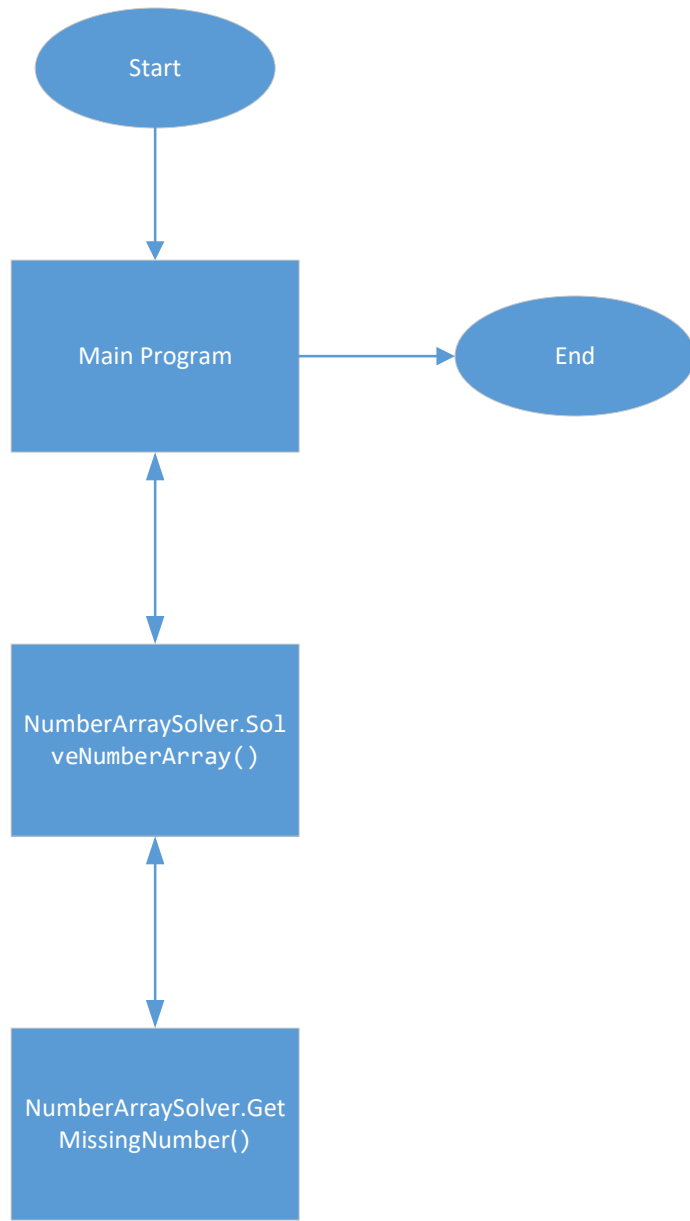
Tests

TestValidNumberList() – Happy path for when there is a list of numbers and only 1 missing.

TestNULLNumberList() – Negative edge case that passes in a null list of numbers and tests that the ArgumentNullException is called.

TestNoItemsInListNumberList() – Negative edge case that passes in a valid list of numbers object but the list has no elements in it. The tests expects the InvalidDataException exception is thrown.

TestMoreThan1MissingNumberNumberList() – Negative edge case that passes in a list of numbers from 1-10 with 2 numbers missing. The ArgumentOutOfRangeException is tested.

Program Flow

In this solution, I created a console program to provide the user output. I created a separate class library for the array solver. I created a test project to run.

NumberArraySolver Class
SolveNumberArray() – this method validates the numberList parameter by testing for NULL, no elements. It then calls the GetMissingNumber method and throws an error message if there is more than one missing number.

GetMissingNumber() – this method increments through the specified number list and compares the current number to the next. If the next value is not n + 1, then the number is missing. This method returns a list of integers.

ArraySolver Class
   This class is the base class of the solver assembly. It's defined as an abstract class and has 1 virtual method named SolveNumberArray. The ArraySolver inherits from the IArraySolver interface. The interface requires any inheriting class to have a method named **SolveNumberArray**.