## Add(T) Complexity

|  |
| --- |
|  |
|  |
|  |

Array.Copy() has complexity O(n), the other lines have O(1). Complexity is always O(n).

## Remove(index) Complexity – Worst Case

Array.Copy() has complexity O(n), the other lines have O(1). Complexity is always O(n).

## Remove(index) Complexity – Best Case

|  |
| --- |
| Array.Copy() has complexity O(n), the other lines have O(1) |
|  |
|  |
| Best Case is when the array has only 1 element. |
|  |
|  |
| Complexity is always O(n) |

## Remove(index) Complexity – Average Case

|  |  |
| --- | --- |
| Array.Copy() has complexity O(n), the other lines have O(1). | |
| Complexity is always O(n). |

## RemoveFirst(T) Complexity

|  |
| --- |
| The first call to Array.Copy performs O(1) operations, the second call of Array.Copy performs O(n) operations. |
|  |
|  |
| Complexity is O(n). |

## RemoveLast(T) Complexity

|  |
| --- |
|  |
|  |
| The first call to Array.Copy performs O(n) operations, the second call of Array.Copy performs O(1) operations. |
|  |
|  |
| Complexity is O(n). |

## Length Complexity

Complexity is O(1) because the array saves its length.

## This[index] Complexity

Complexity is O(1) because the array can calculate the index in a few operations.

## First Complexity

Complexity is O(1) because it is a special case of the indexer.

## Last Complexity

Complexity is O(1) because it is a special case of the indexer.