|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Generic** | **Description** | **Internal Implementation** | **Add**  **insert** | **Add beyond capacity** | **Queue**  **Push** | **Dequeue**  **Pop**  **Peek** | **Remove**  **RemoveAt** | **Item[]**  **ElementAt()** | **GetEnumerator** | **Contains()**  **IndexOf()**  **Find** |
| List | Represents a strongly typed list of objects that can be accessed by index. Provides methods to search, sort, and manipulate lists. | Array | O(1)/O(N) | O(N) | *NA* | *NA* | O(N) | O(1) | O(1) | O(N) |
| LinkedList | Represents a doubly linked list. | Doubly linked list | O(1) | O(1) | O(1) | O(1) | O(1) | O(n) | O(1) | O(n) |
| Stack | Represents a variable size last-in-first-out (LIFO) collection of instances of the same specified type. | Array | O(1) | O(n) | O(1) | O(1) | *NA* | *NA* | O(1) | O(n) |
| Queue | Represents a first-in, first-out collection of objects. | Array | O(1) | O(n) | O(1) | O(1) | *NA* | *NA* | O(1) | O(n) |
| Dictionary | Represents a collection of keys and values where keys can’t be duplicated and can’t be null. | Hashtable with links to another array index for collision | O(1)/O(n) | O(n) | *NA* | *NA* | O(1)/O(n) | O(1)/O(n) | O(1) | O(n) |
| HashSet | Same as Dictionary but with no values | Hashtable with links to another array index for collision | O(1)/O(n) | O(n) | *NA* | *NA* | O(1)/O(n) | O(1)/O(n) | O(1) | *NA* |
| SortedDictionary | Same as Dictionary but sorted on the key and uses a tree implementation for searching | Red-black tree | O(log n) | O(log n) | *NA* | *NA* | O(log n) | O(log n) | O(log n) | O(n) |
| SortedList | Same as List but sorted using a provided compare function and it uses binary search | Array | O(n), O(log n) if added to end of list | O(n) | *NA* | *NA* | O(n) | O(log n) | O(1) | O(n) |
| SortedSet | Same as SortedDictionary but with no value | Red-black tree | O(log n) | O(log n) | *NA* | *NA* | O(log n) | O(log n) | O(log n) | *NA* |