**Question:** : Compare the following aspects of linked lists and dynamic arrays:

**Comparison of Linked Lists and Dynamic Array**

1. Time Complexity of Each Method

|  |  |  |
| --- | --- | --- |
| **Operation** | **Singly Linked List** | **Dynamic Array** |
| Insert at index | O(n) | O(n) |
| Delete at index | O(n) | O(n) |
| Get size | O(1) | O(1) |
| Is empty | O(1) | O(1) |
| Rotate right by k | O(n) | O(n) |
| Reverse | O(n) | O(n) |
| Append | O(1) | O(1) |
| Prepend | O(1) | O(n) |
| Merge | O(1) | O(n) |
| Interleave | O(n) | O(n) |
| Find middle | O(n) | O(1) |
| Split at index | O(n) | O(n) |
| Resize (custom factor) | — | O(n) |
|  |  |  |
|  |  |  |

**2. Space Complexity of Each Method**

|  |  |  |
| --- | --- | --- |
| **Operation** | **Singly Linked List** | **Dynamic Array** |
| Insert at index | O(1) | O(1) |
| Delete at index | O(1) | O(1) |
| Get size | O(1) | O(1) |
| Is empty | O(1) | O(1) |
| Rotate right by k | O(1) | O(1) |
| Reverse | O(1) | O(1) |
| Append | O(1) | O(1) |
| Prepend | O(1) | O(n) |
| Merge | O(1) | O(n) |
| Interleave | O(1) | O(n) |
| Find middle | O(1) | O(1) |
| Index of element | O(1) | O(1) |
| Split at index | O(1) | O(n) |
| Resize (custom factor) | O(1) | O(n) |
|  |  |  |

**3**. Advantages and Disadvantages

**Linked Lists**

**Advantages:**

1. Dynamic Size

2. Efficient Insertions/Deletions

3. No Wasted Space

**Disadvantages:**

1. Slow Access

2. Memory Overhead

3. Cache Performance

**Dynamic Arrays**

**Advantages:**

1. Fast Access

2. Efficient Iteration

3. Memory Efficiency

**Disadvantages:**

1. Resize Overhead

2. Insert/Delete Costs

3. Pre-allocated Space