**Feb 17, 2021**

**List: a basic data structure in computer science**

**One-dimension data structure**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | **Unique**  **predecessor** | **Current**  **element** | **Unique**  **successor** |  |  |  |  |

**List: It can be classified into two groups: unsorted or sorted lists**

**Unsorted List: there is no particular order that exists in the list**

**Sorted List: There is an ascending or descending order in the list**

**How to find a particular element in a list?**

**Search methods:**

1. **Linear search: we search a list from the head to tail one by one in sequence (implemented by a for loop; slow in execution: O(n), where n is the number of elements in the list )**
2. **Binary search: this is a quick way to search an item from a sorted list. Time complexity: O(log n)**

**Example: n = 10^6**

**Log n = log(10^6) = 6**

**Implementation: array or linked list**

**Unsorted List:**

|  |
| --- |
| **Max** |
| **Bradley** |
| **Asad** |
|  |
|  |

**Add: John into the list**

1. **Add the item to the first empty spot**
2. **Length++**

|  |
| --- |
| **Max** |
| **Bradley** |
| **Asad** |
| **John** |
|  |

**Remove Bradley from the list (algorithm)**

1. **Find the location of the item to be deleted**
2. **Copy the last element to the item to be deleted**
3. **Length—**

**//typedef int ItemType;**

**typedef float ItemType;**

**Sorted List:**

|  |
| --- |
| **Asad** |
| **Bradley** |
| **Maxwell** |
| **Tom** |
|  |

**Add: John into the list**

1. **Find a location to insert (linear or binary search)**
2. **Shift all the remaining elements downward by one position**
3. **Insert the item to the location**
4. **Length++**

|  |
| --- |
| **Asad** |
| **Bradley** |
| **John** |
| **Maxwell** |
| **Tom** |

**Delete an item from a sorted list:**

|  |
| --- |
| **Asad** |
| **Bradley** |
| **John** |
| **Maxwell** |
|  |

**Delete Bradley,**

1. **Find the location of the item to be deleted**
2. **Move the remaining list upward by one position**
3. **Length--**

|  |
| --- |
| **Asad** |
| **John** |
| **Maxwell** |
| **Maxwell** |
|  |

**Time Complexity:**

**O( ) notation**

**For(int i=0; i<n; i++)**

**For(int j=0; j<n; j++)**

**X[i][j] = i\*j;**