Assignment 2   
Advanced Programming

**ArrayList vs Vector**

|  |  |
| --- | --- |
| **ArrayList** | **Vector** |
| Multiple threads can use at the same time | One thread at a time |
| Extends array size by 50% | Extends array size by 100% i.e doubles |
| Fast because not synchronized | Slower because synchronized |
| Traverse elements using Iterator | Traverse elements using Enumeration |

**HashSet Vs SortedSet**

|  |  |
| --- | --- |
| **Hashset** | **SortedSet** |
| Not sorted | sorted |
| Uses hashset | Uses treeset |
| Hashing used so O(1) | Binary search as tree used O(log n) |
| Contiguous storage | No contiguous storage |

**TreeSet vs HashSet**

|  |  |
| --- | --- |
| **TreeSet** | **HashSet** |
| Uses tree balanced tree | Uses hashing |
| Doesn’t allow null element | Allows null value |
| Multiple threads can access at a time | Multiple threads can access at a time |
| Implements NavigableSet | Extends AbstractSet |

**Array Vs List**

|  |  |
| --- | --- |
| **Array** | **List** |
| Static Memory Allocation(Stack) | Dynamic Memory Allocation(Heap) |
| Fixed Size | Variable Size |
| No Iterator (default) | Iterator available |
| No remove or add facility (just overwrite or add if space available) | Remove, Add |

**NavigableSet vs NavigableMap**

|  |  |
| --- | --- |
| **NavigableSet** | **NavigableMap** |
| collection | Map |
| Values | Key-Value pairs (distinct Keys) |
| Uses treeSet | Uses treeMap |
|  |  |