**Table of data structures – Advantages and disadvantages**

| **Data Structure** | **Advantages** | **Disadvantages** |
| --- | --- | --- |
| **Array** | Quick inserts Fast access if index known< | Slow search Slow deletes Fixed size |
| **Ordered Array** | Faster search than unsorted array | Slow inserts Slow deletes Fixed size |
| **Stack** |  |  |
| **Queue** |  |  |
| **Linked List** | Quick inserts Quick deletes | Slow search |
| **Binary Tree** | Quick search Quick inserts Quick deletes *(If the tree remains balanced)* | Deletion algorithm is complex |
| **Red-Black Tree** | Quick search Quick inserts Quick deletes *(Tree always remains balanced)* | Complex to implement |
| **2-3-4 Tree** | Quick search Quick inserts Quick deletes *(Tree always remains balanced)* *(Similar trees good for disk storage)* | Complex to implement |
| **Hash Table** |  | Slow deletes Access slow if key is not known Inefficient memory usage |
| **Heap** |  | Slow access to other items |
| **Graph** |  | Some algorithms are slow and very complex |