# Comparison of Data Structures in C++, Python, and Java

This document provides a side-by-side comparison of commonly used data structures in C++, Python, and Java. It includes details on built-in libraries and their features.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Data Structure | C++ (STL & Native) | Python (Built-in) | Java (Collections Framework) | Key Differences/Notes |
| Array | std::array, std::vector | list | Array, ArrayList | Python's list is dynamic; Java's Array has fixed size, ArrayList is dynamic. |
| Dynamic Array | std::vector | list | ArrayList | std::vector and ArrayList have similar behavior with dynamic resizing. |
| Linked List | std::list (Doubly Linked List) | Custom implementation | LinkedList | C++ and Java provide built-in linked lists; Python requires manual implementation. |
| Stack | std::stack, std::vector | list or collections.deque | Stack, Deque | Python lacks a dedicated stack class; list or deque is used. |
| Queue | std::queue, std::deque | collections.deque | Queue, LinkedList, Deque | C++ and Java have dedicated queue interfaces; Python uses deque. |
| Deque (Double-Ended Queue) | std::deque | collections.deque | Deque | All languages offer efficient implementations of deque. |
| Priority Queue | std::priority\_queue | heapq | PriorityQueue | Python's heapq is a module, not a class; C++ and Java have dedicated classes. |
| Hash Table | std::unordered\_map | dict | HashMap, Hashtable | All languages support hash tables; Java's Hashtable is synchronized. |
| Set | std::unordered\_set, std::set | set | HashSet, TreeSet | Python's set is hash-based; Java's TreeSet provides sorted elements. |
| Map/Dictionary | std::map, std::unordered\_map | dict | HashMap, TreeMap | Python uses dict; Java has sorted TreeMap for ordered key-value pairs. |
| Graph | Custom implementation | Custom implementation | Custom implementation | Graphs are not built-in but can be implemented using lists/maps. |
| Heap | std::priority\_queue | heapq | PriorityQueue | Python and Java support min-heaps by default; C++ provides max-heap by default. |
| String | std::string | str | String, StringBuilder | All languages support string manipulation, but C++ requires manual effort for safety. |
| Bitset | std::bitset | Custom (list, int manip.) | Custom (BitSet class) | C++ has a dedicated bitset; Python and Java rely on custom approaches. |
| Tree | Custom implementation | Custom implementation | Custom implementation | Trees need to be implemented manually or with third-party libraries in all languages. |