

C++

Sorting & Big-O Complexity — Algorithms

* You are here.

ADTs

Lists

Linked Lists

⋮

} General purpose
data structures,
algorithms, classifications

Arrays

- + Many elements of same type
- + Get by index in $O(1)$
- + Set by index in $O(1)$
- + Compact

- no insert
- can't change size
- homogeneous

- no index checking
- no knowledge of size

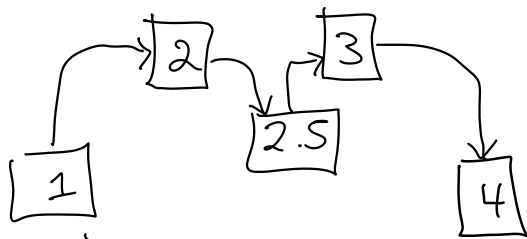
List

index
checking

- void insertFront(string value) | void insertBack(string value)
- int getSize()
- void removeFront() | void removeBack()
- string get(int index)

Abstract Data Type (e.g. List)

LinkedList is a kind of List
↓ concrete data structure ↓ ADT



```
template <typename T>
class LinkedListNode {
public:
    T value;
    LinkedListNode* next; // last box will have nullptr
};
```

```
template <typename T>
class LinkedList : public List<T> {
public:
    ...
private:
    LinkedListNode<T>* head;
};
```

Method getSize():

Method getSize()

node ← this→head

count ← 0

While node ≠ null:

node ← node→next

count ← count + 1

End While

Return count

End Method

$O(n)$



node

count 3

For faster getSize, add a field "size" starts at 0 changes w/ chain

$O(1)$

Invariant : the field "size" always contains length of chain