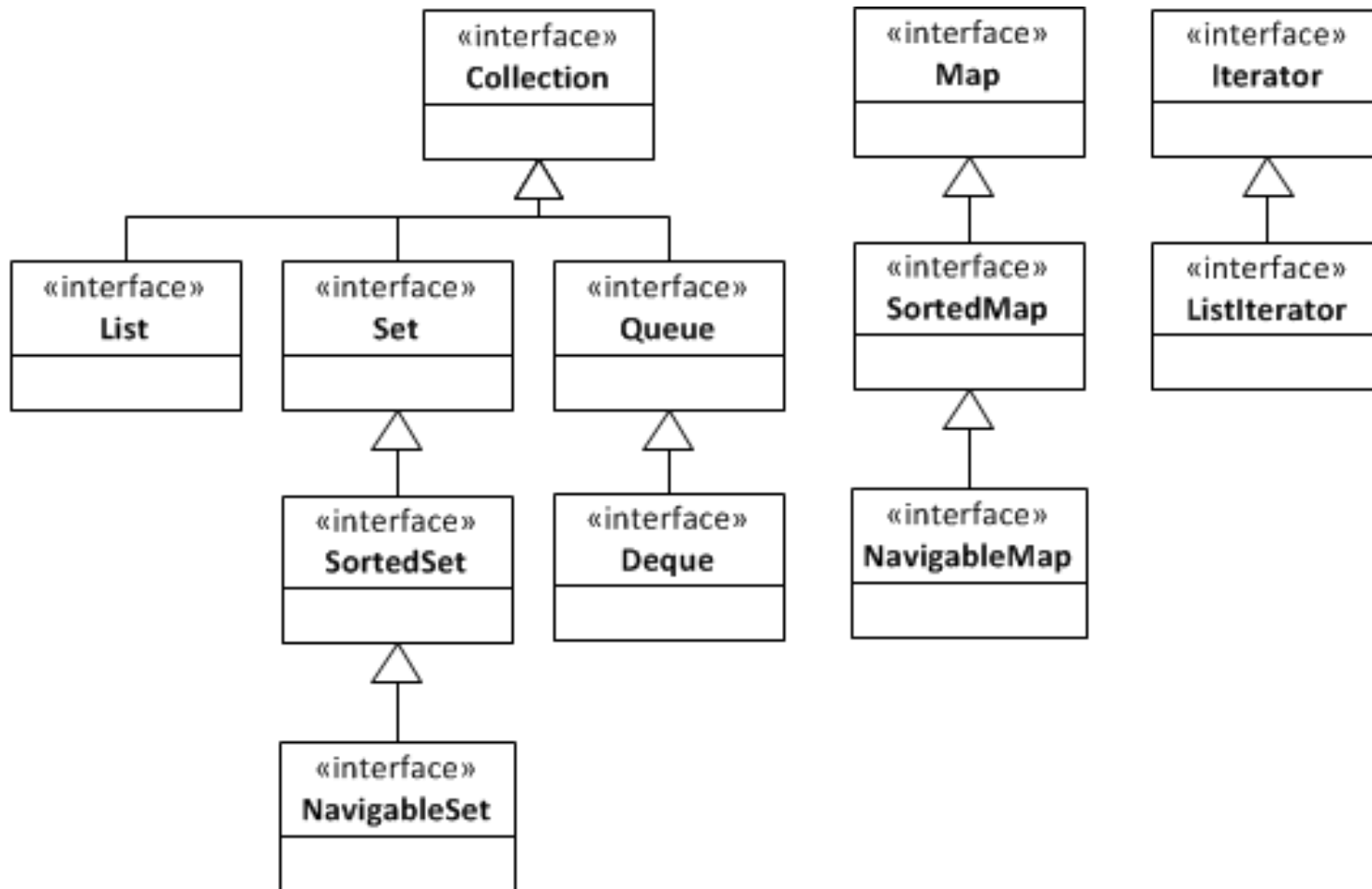


# Java Collections

CS 240 – Advanced Programming Concepts

# Java Collections

- [Collection](#), [Iterator](#), [Algorithms](#) (Collections Class)



# List

- [List interface](#)
- A sequence of elements accessed by index  
get(index), set(index, value)
- ArrayList (resizable array implementation)
- LinkedList (doubly-linked list implementation)

# Set

- [Set interface](#)
- A collection that contains no duplicates  
add(value), contains(value), remove(value)
- HashSet (hash table implementation)
- TreeSet (bst implementation)
- LinkedHashSet (hash table + linked list impl)

# Queue

- [Queue interface](#)
- A collection designed for holding elements prior to processing  
add(value), peek(), remove()
- ArrayDeque (fifo, resizable array impl)
- LinkedList (fifo, linked list implementation)
- PriorityQueue (priority queue, binary heap impl)

# Deque

- [Deque interface](#)
- A queue that supports efficient insertion and removal at both ends
  - addFirst(value), addLast(value),
  - peekFirst(), peekLast(),
  - removeFirst(), removeLast()
- ArrayDeque (resizable array implementation)
- LinkedList (linked list implementation)

# Stack

- Java's Stack class is deprecated
- If you need a stack, use a Deque
  - push() => Deque.addFirst()
  - pop() => Deque.removeFirst()
  - peek() => Deque.peekFirst()

# Map

- [Map interface](#)
- A collection that maps keys to values
  - A set of (key, value) pairs where keys are unique

put(key, value), get(key), contains(key), remove(key)  
keySet(), values(), entrySet()
- HashMap (hash table implementation)
- TreeMap (bst implementation)
- LinkedHashMap (hash table + linked list impl)