Deques

- Double Ended Queues
- Supports insertion and deletion at both end of the queue

Examples:

- Resturant waiting list, add someone, then realize their table isnt availbe

The Deque Abstract Data Type

D.add_first(e): Add element e to the front of deque D.

D.add_last(e): Add element e to the back of deque D.

D.delete_first(): Remove and return the first element from deque D; an error occurs if the deque is empty.

D.delete_last(): Remove and return the last element from deque D; an error occurs if the deque is empty.

D.first(): Return (but do not remove) the first element of deque D; an error occurs if the deque is empty.

D.last(): Return (but do not remove) the last element of deque D; an error occurs if the deque is empty.

D.is_empty(): Return True if deque D does not contain any elements.

len(D): Return the number of elements in deque D; in Python, we implement this with the special method `__len__`.

Running Time of Deque

- All operations having a O(1) running time

Deques in the Python Collection Module

Our Deque ADT	collections.deque	Description
len(D)	len(D)	number of elements
D.add_first( )	D.appendleft( )	add to beginning
D.add_last( )	D.append( )	add to end
D.delete_first( )	D.popleft( )	remove from beginning
D.delete_last( )	D.pop( )	remove from end
D.first( )	D[0]	access first element
D.last( )	D[−1]	access last element
	D[j]	access arbitrary entry by index
	D[j] = val	modify arbitrary entry by index
	D.clear( )	clear all contents
	D.rotate(k)	circularly shift rightward k steps
	D.remove(e)	remove first matching element
	D.count(e)	count number of matches for e

Table 6.4: Comparison of our deque ADT and the collections.deque class.

Example Python Code:

```
import collections

# Create a deque
DoubleEnded = collections.deque(["Mon","Tue","Wed"])
print (DoubleEnded)

# Append to the right
print("Adding to the right: ")
DoubleEnded.append("Thu")
print (DoubleEnded)

# append to the left
print("Adding to the left: ")
DoubleEnded.appendleft("Sun")
print (DoubleEnded)

# Remove from the right
print("Removing from the right: ")
DoubleEnded.pop()
print (DoubleEnded)

# Remove from the left
print("Removing from the left: ")
DoubleEnded.popleft()
print (DoubleEnded)

# Reverse the dequeue
print("Reversing the deque: ")
DoubleEnded.reverse()
print (DoubleEnded)
```

Returned:

```
deque(['Mon', 'Tue', 'Wed'])
Adding to the right:
deque(['Mon', 'Tue', 'Wed', 'Thu'])
Adding to the left:
deque(['Sun', 'Mon', 'Tue', 'Wed', 'Thu'])
Removing from the right:
deque(['Sun', 'Mon', 'Tue', 'Wed'])
Removing from the left:
deque(['Mon', 'Tue', 'Wed'])
Reversing the deque:
deque(['Wed', 'Tue', 'Mon'])
```