

## Coming up:

*Due Fri, Jan 26: (end of the day)*

*Read (or finish reading ) Section 2.(2, 4, & 5)*

*Take data structures quiz*

*Also:*

*Do “**Implementing ADTs**” project (due Mon, Jan 29)*

*Due Wed, Jan 31: (class time)*

*Read Section 3.1*

*Do Exercises 2.(22–24)*

*Due Thurs, Feb 1: (end of day) Take sorting quiz*

This week and next week (Chapters 2 and 3):

- ▶ Abstract data types (Monday)
- ▶ Data Structures (**today** and Friday)
- ▶ Linear time sorting (next week Monday and Wednesday)

Today:

- ▶ Ex 1.11
- ▶ ADT review
- ▶ Data structure categories
- ▶ List vs array
- ▶ Abstractions
- ▶ Adapter pattern

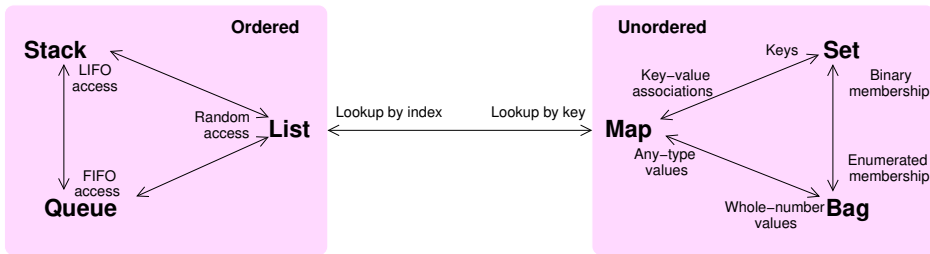
```
def is_palindrome(str) :  
    palindromic = True  
    n = len(str)  
    i = 0  
    while palindromic and i < n // 2 :  
        palindromic = str[i] == str[n-i-1]  
        i += 1  
    return palindromic
```

Invariant (Loop of is\_palindrome)

1.  $\forall j \in [0, i - 1), \text{str}[j] = \text{str}[n - j - 1]$
2. *palindromic* iff ( $i = 0$  or  $\text{str}[i - 1] = \text{str}[n - i]$ )
3.  $i$  is the number of iterations completed

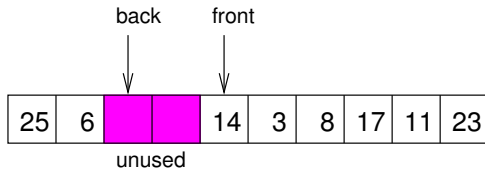
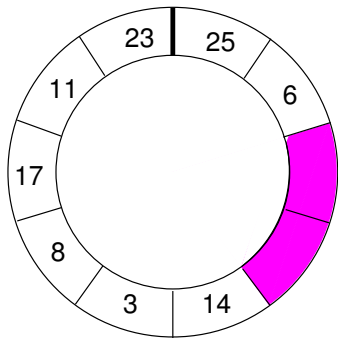
The “canonical ADTs”:

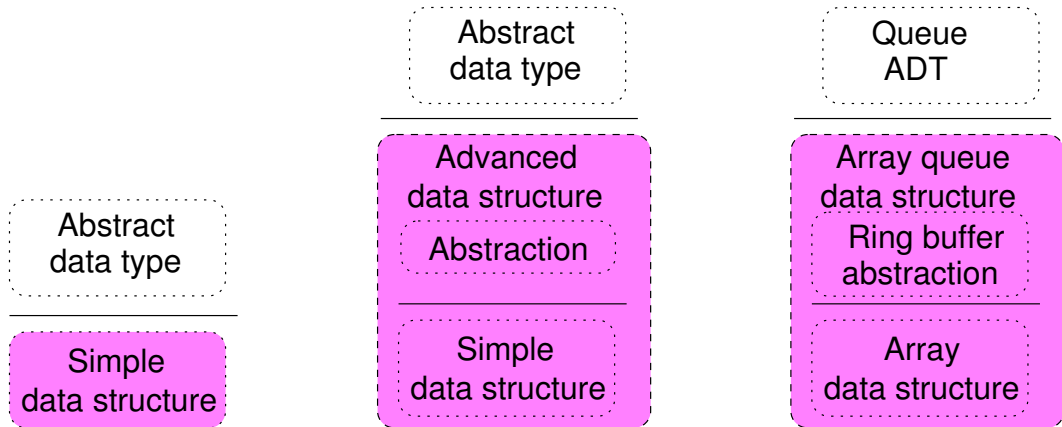
- List.** Linear collection with sequential and random access.
- Stack.** Linear collection with LIFO access.
- Queue.** Linear collection with FIFO access.
- Set.** Unordered collection with binary membership.
- Bag.** Unordered collection with enumerated membership.
- Map.** Unordered collection of associations between keys and values.



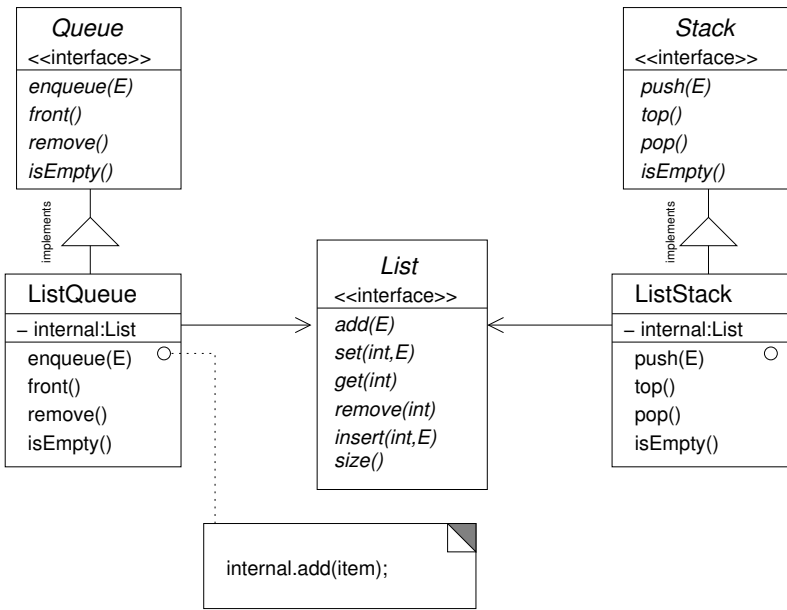
The four basic ways to implement an ADT:

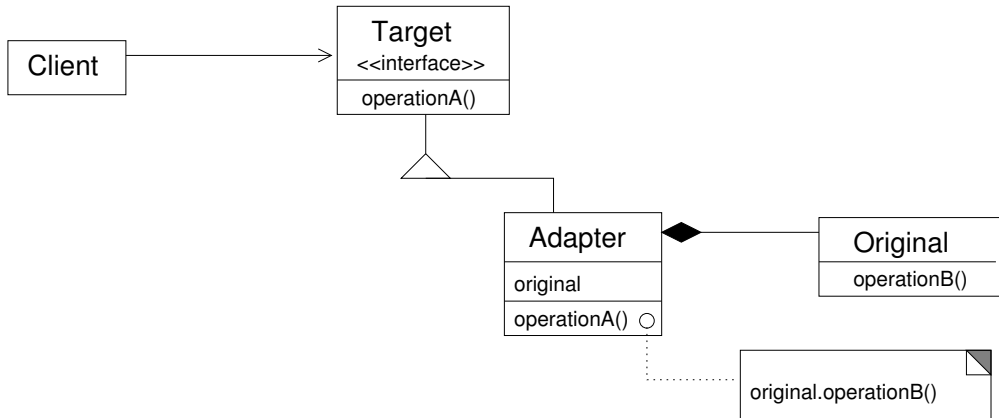
- ▶ Use an array
- ▶ Use a linked structure
- ▶ Use an “advanced” data structure, varying and/or hybridizing linked structures and arrays
- ▶ Adapt an existing implementation of another ADT.

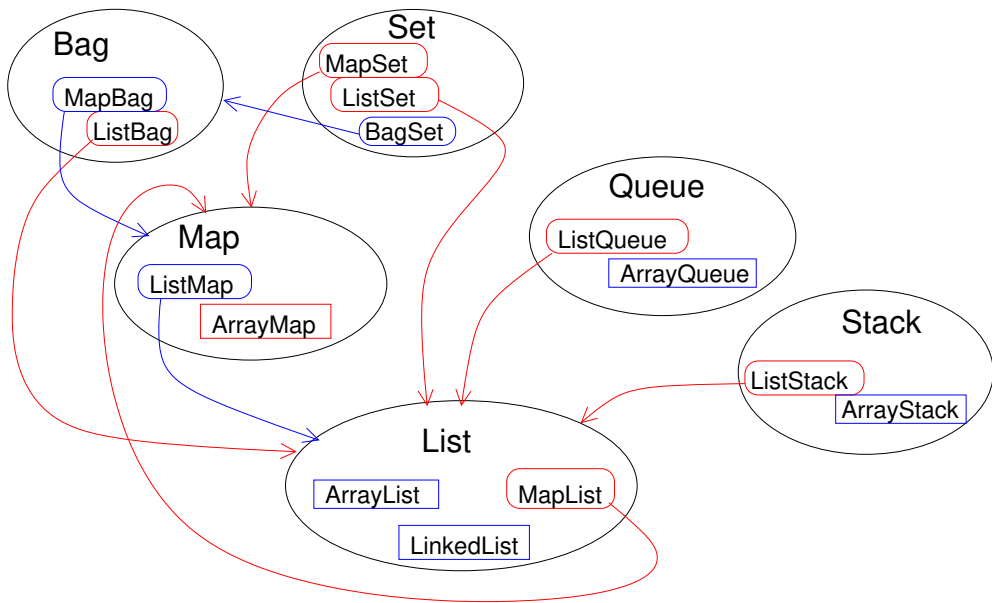












## Coming up:

*Due Fri, Jan 26: (end of the day)*

*Read (or finish reading ) Section 2.(2, 4, & 5)*

*Take data structures quiz*

*Also:*

*Do “**Implementing ADTs**” project (due Mon, Jan 29)*

*Due Wed, Jan 31: (class time)*

*Read Section 3.1*

*Do Exercises 2.(22–24)*

*Due Thurs, Feb 1: (end of day) Take sorting quiz*