Stacks

1. Think of a stack of dishes at a Buffet
2. Can only access the top dish
3. Can only access top value
4. Last In First Out (LIFO)
   1. Can only access the top value
5. Push – items into the stack
6. Pop – items off the top of the stack
7. Top – shows the item on the top
8. Can hold any element that you want
9. Custom classes
   1. Must define operators <, >, ==
10. Use when
    1. You want to reverse a process
       1. Push process onto the stack, the pop until empty
    2. You want to hold onto the precise order of something
    3. Need to access newer elements before older elements

Queue

1. Line
2. First in First Out (FIFO)
3. Push – places item into the queue
4. Pop – takes from the queue
5. Front – item at the beginning of the queue
6. Use when
   1. You want a FIFO sequence
   2. Line of customers
   3. Items done in a specific order
   4. Sequences
7. Priority queue
   1. Higher priority items moved to front of queue

Set

1. Collection of unique objects
2. Associative, referenced by a key
   1. Key is the index
   2. O(1) for most lookup, sometime O(log n) – depends on language and implementation
3. Really good for searching by the key
4. Automatically store in order
   1. set <datatype, less <string>> //store in ascending order
   2. set <datatype, greater <string>> //store in descending order
5. Must have the comparator built in the class
   1. <, >, ==