CS 61B!

1. When am I useful Senpai?

1. Customer in a line ⇒ first in first out (FIFO)

⇒ Queue (D)

2. Requirements: fast for { insert, delete } O(1)

not fast for { search, lookup } O(n)

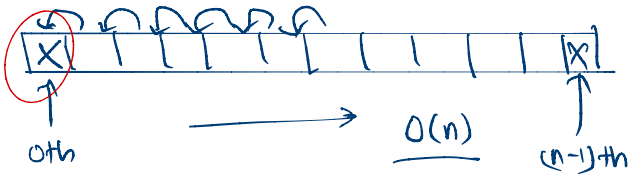
Linked List (B)

3. Constant time access

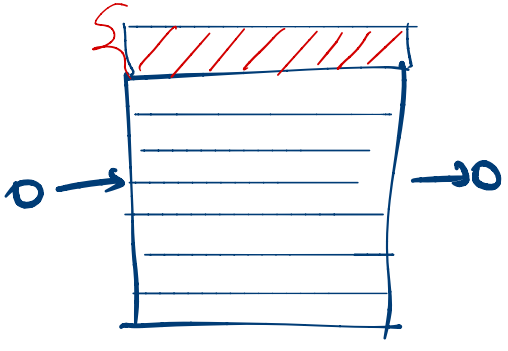
Arrays (A) → assume in 61B arr[i] is Θ(1)

4. Last in First Out (LIFO)

Stack

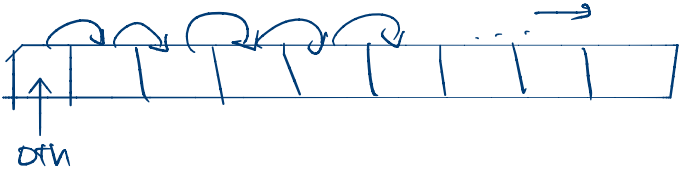


n elts



push() — add to the top

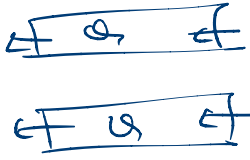
pop() — remove from top

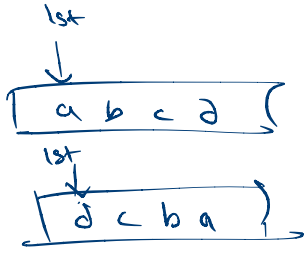


2. Reverse Me

Input:

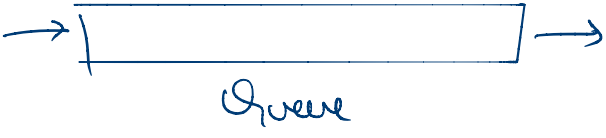
Output:

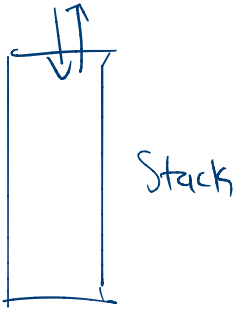


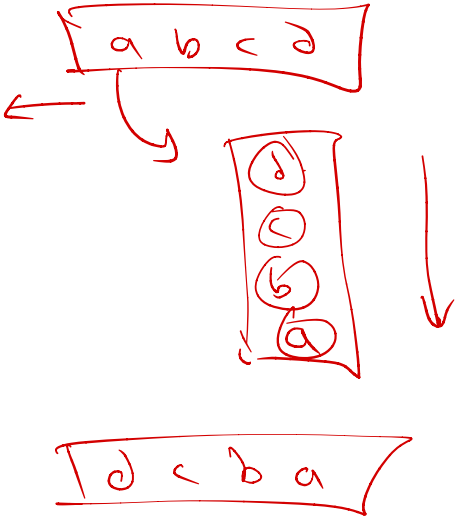



```
private static void reverse ( MyIntQ q) {
    Stack s = new Stack(); // make a stack
    while ( ! q.isEmpty()) {
        s.push( q.dequeue()); //
    }
→   while (!s.isEmpty()) {
        q.enqueue (s.pop());
    }
}
```

s.push (q.dequeue())
removes first elt of q
adds it to s



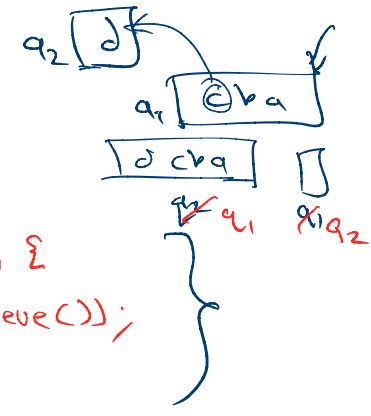




FIFO — Q

LIFO — S

3. Pseudo Stack

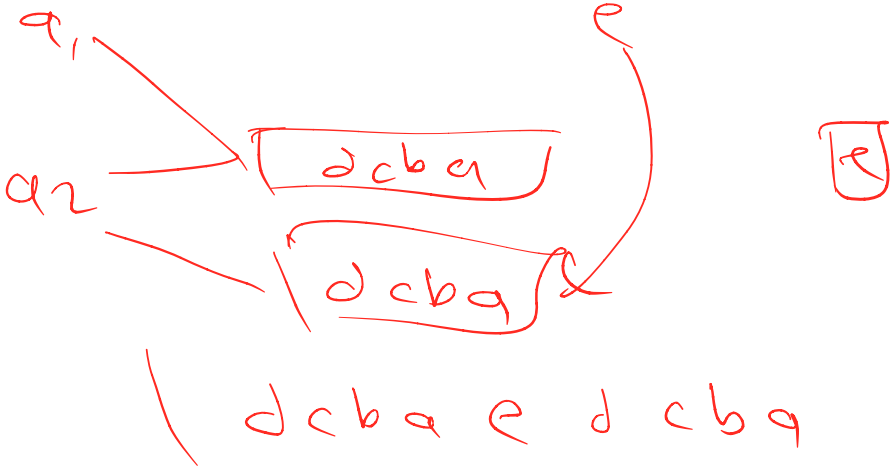



```
public void push (int item) {
    q2.enqueue (item);
    while (!q1.isEmpty()) {
        q2.enqueue (q1.dequeue());
    }
    MyIntQueue tempQ = q1;
    q1 = q2;
    q2 = tempQ;
}
```

q_1 → contains the elts in reverse order

```
public int pop () {
    if (!q1.isEmpty()) {
        return q1.dequeue();
    } else {
        // throw Exception
    }
```

Figure out how to add the latest element to front



---

push	pop
add to front of queue	remove from queue

4. A Balancing Act

str.charAt(i);

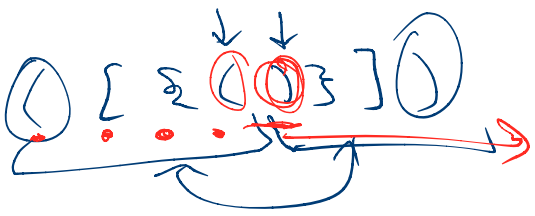
```
private static boolean hasValidParens(String str) {
    Stack s = new Stack();
    for(int i=0; i<str.length(); i++) {
        char c = str.charAt(i);
        if(c=='{' || c=='(' || c=='[') {
            s.push(getRightParen(c));
        } else {
            if(s.isEmpty()) {
                return false;
            }
            if(c!=s.pop()){
                return false;
            }
        }
    }
    return s.isEmpty();
}
```

// F ⇒ more close than open

// F ⇒ order parens is wrong (mismatch)

// F ⇒ More open than close

// T ! found a valid string



very last
Character for opening

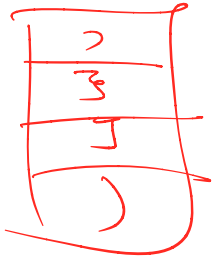
= very first
for closing

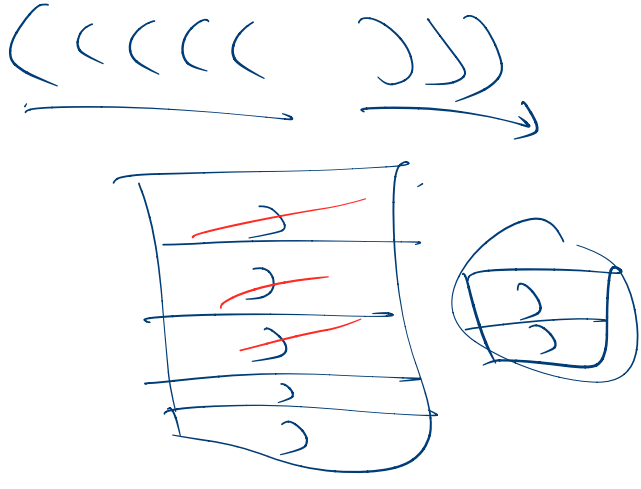
getRightParen (char)

'{' → '}'

Stack stores opening chars

LIFO





```
if ( ——— )
    ret true;
else
    ret false;
```

ret ———;