JAVA DEQUES
A library is a collection of frequently used tools to facilitate programming large applications (or other libraries).

Examples you probably have seen:
- `java.Math`
- `java.util.Scanner`
- `java.util.Random`

Other examples:
- Access to servers/databases
- Graphics
- Reflection ("self-aware" classes)
“Truly knowing a language requires knowing the library”

“Libraries are languages”
SUMMARY OF CLASSES
CONCERNING STACKS, QUEUES, AND DEQUES

• **Stack\<E\>** - Java documentation says to avoid (sort of deprecated)

• **ArrayDeque\<E\>** - Growable-array using doubling strategy (supports Deque, Stack, and Queue)

• Others outside the scope of this course

• To find how to use them, go to the Java API!
**EXAMPLE OF USING ARRAYDEQUE<>*E>*\**

1. `Scanner s = new Scanner(new File("numbers.txt"));`
2. `ArrayDeque<Integer> numbers = new ArrayDeque<Integer>();`
3. `while (s.hasNextInt())`
4.   `numbers.add(s.nextInt());`
5.   `...elsewhere...`
6. `int sum = 0;`
7. `for(int n = 0; n < numbers.size(); ++n)`
8.   `sum += numbers.poll(n);`
PROBLEMS

• Rolling average (used to compute statistics for window of streaming data, e.g., miles-per-hour in a car)

• Create a stream of numbers using Brownian noise between 0 and 100 (representing speed of a car)
  • Brownian noise is a random walk. Essentially, randomly move up or down one step (±0.2)

• Use this to compute the rolling average of the last 100 numbers. Continually output the rolling average to the terminal for 100,000 iterations.