Today’s Topics

- **Reading:**
  - Sedgewick & Wayne: 1.1-1.4, 2.1-2.2

- **Topics**
  - APTs
  - Basic Collections:
    - ArrayLists: the expandable array
    - Sets: the list of distinct elements

- **Acknowledgements**
  - Slides from Sedgewick & Wayne

---

Tips for Excelling in CompSci 100e

- Read the Book
- Ask questions
- Keep working until it is correct
- Seek help when stuck
- Visit the professor, TA, and UTAs
- Start early!
- Get the easy points

---

Functions (Static Methods)

- **Java function.**
  - Takes zero or more input arguments.
  - Returns one output value.

- **Applications.**
  - Scientists use mathematical functions to calculate formulas.
  - Programmers use functions to build modular programs.
  - You use functions for both.

- **Examples.**
  - Book libraries `StdDraw.show()`, `StdAudio.play()`.
  - User-defined functions: `main()`.

---

Anatomy of a Java Function

- **Java functions.** Easy to write your own.

```
public static double sqrt(double c) {
    if (c < 0) return Double.NaN;
    double err = 1e-15;
    double t = c;
    while (Math.abs(t - c/t) > err * t) {
        t = (c/t + t) / 2.0;
    }
    return t;
}
```
Libraries

- **Library.** A module whose methods are primarily intended for use by many other programs.

- **Client.** Program that calls a library.

- **API.** Contract between client and implementation.

- **Implementation.** Program that implements the methods in an API.

Algorithmic Programming Testing

- **Algorithmic Programming Testing problems (APTs)**
  - Assigned most weeks, due on Tuesdays
  - Similar to Unit Testing, part of Agile Development

- **Create a single class that matches the definition**
  - Can add whatever other methods you like

- **Graded for correctness**
  - Efficiency matters only in a binary way
  - Good design will be useful to you

What can you put into an ArrayList?

- **Any Object**
  
  ```java
  Scanner in = ...;
  ArrayList<String> list = new ArrayList<String>();
  while (in.hasNext())
  list.add(in.next());
  ```

- **Use a wrapper class (see java.lang.*)**
  - Integer, Double, Character, Boolean,

- **Can have your cake and eat it too**
  
  ```java
  ArrayList<Integer> list = new ArrayList<Integer>();
  for (int k = 0; k < 10; k++)
  list.add(k*k);
  for (Integer jj : list)
  System.out.println(jj);
  ```

- **All made practical by Version 5 of Java**

Sets

- **Set is an unordered list of items**
  - Items are unique! Only one copy of each item in set!

- **We will use two different implementations of sets**

- **TreeSet**
  - A TreeSet is backed up by a tree structure (future topic)
  - Keeps items sorted (+)
  - Slower than HashSets ?? (-)

- **HashSet**
  - A HashSet is backed up by a hashing scheme (future topic)
  - Items not sorted – should seem to be in random order (-)
  - Faster than TreeSets ?? (+)