

# CSE 12 – Basic Data Structures and Object-Oriented Design

## Lecture 2

Greg Miranda & Paul Cao, Winter 2021

This lecture is being recorded

# Announcements

- Quiz 2 due Friday @ 8am
- Survey 1 due Friday @ 11:59pm
- PA1 released today – open collaboration

# Topics

- Add junit to Eclipse project
- Questions on Lecture 2?
- Practice writing tests

# sumNumbers

- Given a string, return the sum of the numbers appearing in the string, ignoring all other characters. A number is a series of 1 or more digit chars in a row. (Note: `Character.isDigit(char)` tests if a char is one of the chars '0', '1', .. '9'. `Integer.parseInt(string)` converts a string to an int.)
- Given a string, return the sum of the numbers appearing in the string, ignoring all other characters. A number is a series of 1 or more digit chars in a row. (Note: `Character.isDigit(char)` tests if a char is one of the chars '0', '1', .. '9'. `Integer.parseInt(string)` converts a string to an int.)

- `sumNumbers("abc123xyz")` → 123
- `sumNumbers("aa11b33")` → 44
- `sumNumbers("7 11")` → 18

**What test cases should we write to confirm that our implementation works?**

# evenOdd

- Return an array that contains the exact same numbers as the given array, but rearranged so that all the even numbers come before all the odd numbers. Other than that, the numbers can be in any order. You may modify and return the given array, or make a new array.
- `evenOdd([1, 0, 1, 0, 0, 1, 1]) → [0, 0, 0, 1, 1, 1, 1]`
- `evenOdd([3, 3, 2]) → [2, 3, 3]`
- `evenOdd([2, 2, 2]) → [2, 2, 2]`

**What test cases should we write to confirm that our implementation works?**