

# Recitation 5

James Wei

Professor Peck

9/27/2013

# Covered in this Recitation

- Intro to JUnit!
- LinkedList Practice!
- No google form—submit through Ambient this week!

# Quick Announcement

- Exam is next Wednesday (10/2)!
- Last minute review session being held next Tuesday (10/1) at 7:30pm in Love Auditorium (usual lecture room)
- Review session will cover important topics for the exam

# First Steps

- Snarf the Recitation code for today
- You should be snarfing two classes:
  - StringLinkedList – we will write methods for this
  - StringLinkedListTester – we will use this to test our implementation
- Take a brief look over the StringLinkedList class, and be sure that you understand how it's working

# Quick Intro to JUnit

- Unit testing framework for Java
- Will run a series of unit tests specified by a tester class and display results in the JUnit view
- Note: if you snarf this code and all of your JUnit calls have errors (could not be resolved) even after importing JUnit, follow these steps:
  - 1) Right click your project in the explorer sidebar
  - 2) Go to “Build Path” -> “Configure Build Path”
  - 3) Click on the “Libraries” tab
  - 4) Click the “Add Library” button on the right sidebar
  - 5) Select “JUnit” from the list and hit next
  - 6) Select “JUnit 4” from the dropdown menu and hit Finish
- If that doesn't fix it then ask the TA separately

# How to use JUnit

- How to set up a tester class:
  - 1) Import junit Test class (org.junit.Test)
  - 2) Create a public void method that takes no argument, e.g. “public void testMethod()”; convention is to prefix test methods with “test”
  - 3) Annotate the method with @Test
  - 4) Add assert() methods to the test method; commonly used ones include assertEquals(), assertNotNull(), assertTrue(), etc. (check javadoc for full list)
  - 5) Repeat steps 2-4 for each test method you want to create
  - 6) Running the tester class will bring up the JUnit sidebar. Each annotated test method will appear as a line item, and those that fail will display a failure trace below which shows exactly which assert statement failed

# Couple Notes about JUnit

- Will be used in assignments later on in the semester (DNA, Boggle)
- Will NOT be tested on
- This knowledge is for your own benefit so that you are aware of the tools you have available to you as you code in the future

# StringLinkedListTester

- First we will familiarize ourselves with the testing tool
- Open StringLinkedListTester.java and run it
- Five tests fail:
  - testAddAtEnd
  - testCompareTo
  - testReverse
  - moveToEnd
  - testDouble
- We will fix all of these methods (yay!)



# StringLinkedList

- Let's code addAtEnd together!
- Complete the rest of the methods on your own, so that the JUnit tests all succeed—write the code on paper first as an exam prep exercise!
- Ask the TAs if you have any questions
- Submit your code via Ambient when you're done!

# Have a good weekend!

Don't forget to submit your code!