

# CSE 12

DI-3

Project 2/ Junit

Vineet Kumar  
vkumar@cs

- Project 1 Feedback/ Comments
- Project 2 – Doubly Linked List with generics;  
and focus on unit testing

# Unit Testing

- Unit testing means to test individual components of your program
- Junit is useful as it allows you to unit test by using an 'Assertion' feature
- Thus you can see what you expected and what you got to test your program
- Helps you to write test cases quickly

# Using Junit

- Download junit from the project2 page
- Use only junit4.8.2 version – Note that for consistency we will only be using this version!

# Writing a Simple test case

- Let us say your class looks like this
  - Class MyNum {  
    int val;  
    public MyNum add(MyNum m) { .....}
- Now, you want to unit test the add function

- You can do this using Junit

```
@Test public void testmyadd() {  
    MyNum num1 = new MyNum(100);  
    MyNum num2 = new MyNum(200);  
  
    MyNum expected = new MyNum(300);  
    MyNum result    = num1.add(num2);  
  
    assertTrue(expected.equals(result));  
}
```

- Isn't that simple!
- But you might want to just write the assert statements and reuse the test data you created
- Use fixtures

```
@Before public void setup()
```

```
{
```

```
    // Add your data initialization code here
```

```
}
```

# How do you compile and run with Junit

- Compiling

- First download the jar from [junit.org](http://junit.org). NOTE the version number!

```
javac -cp ./path/junit4.8.2.jar mytest.java
```

- Running

```
java -cp ./path/junit4.8.2.jar  
org.junit.runner.JUnitCore testclass
```



# What else can you test?

- How about exceptions
  - Remember from PA1 – your methods threw exceptions when say you were trying to remove from an empty list
  - How do you test that?
  - Expected exceptions!  
Try an operation that will always fail  
`list.get(10) // when there are say just 2 elements`

- try {

```
list.get(10);
```

```
assertTrue(false); // U shld not reach here.
```

```
} catch (IndexOutOfBoundsException).....
```