

# 15-121 Data Structures

---

## Introductions

- myself
- the course
- web page

Class web page is in your Blackboard

[F15-Introduction to Data Structures](#)

# Who am I?

---

## ■ Jacobo Carrasquel

- Carnegie Mellon, Faculty at SCS
- Advisor for CS freshman
- [www.cs.cmu.edu/~jxc](http://www.cs.cmu.edu/~jxc)
- How to find me
- [jacobo@cs.cmu.edu](mailto:jacobo@cs.cmu.edu)
- (412)268-6729, GHC 6019
- my secretary ,
  - Angie, [angieb@cs.cmu.edu](mailto:angieb@cs.cmu.edu), (412) 268-5483

# Course goals

---

- learning useful Object-Oriented concepts & techniques
- Learning about common data structures and algorithms in computer science
  - Specifically Java Collections

# What you need

---

## ■ software

- Java SE, Development Kit 8u60 (free download)
  - read pdf doc
  - <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>
- Eclipse Java Classic 4.3 M6 (free download)
  - read pdf doc
  - <http://www.eclipse.org/downloads/packages/eclipse-classic-43-m6/keplerm6>

## ■ Textbook (suggested, not required!)

- *Data Structures*, 3rd Edition, by Koffman & Wolfgang

# Course Structure

---

- Homeworks (15%)
  - Programming assignments
- Weekly Quizzes (25%)
  - written
- Exams (40%)
  - 2 written, closed book, closed notes
- Final Exam (20%)
  - written, closed book, closed notes
- There is no curve applied to any of these grades!

# Homeworks duedates

---

- ALL homeworks are due by 11:50 pm on Tuesdays.
  - You have three(3) days to use up for lateness
  - You can only use up to one(1) late day per homework

If you get seriously ill you must present a written medical excuse from a hospital/doctor as soon as possible!

# Testing your solution

---

- We provide basic testing of your code:
  - Testing is done online, automatically
  - You need to submit your solution (Autolab)
  - You can do this a maximum of FIVE (5) times!

# To submit your solution

---

- We are using a system named Autolab
- Download the ZIP file from web site for the class
  - Each homework has a template (code)
- Edit the necessary source files (java files)
- Create a .ZIP file with your edited files
- Go to <https://autolab.andrew.cmu.edu>
  - demo this Wednesday in recitation



# Important URLs

---

- Class web page
  - Login to Blackboard
  - [F15-Introduction to Data Structures](#)
- To submit your solutions
  - <https://autolab.andrew.cmu.edu>
- [Piazza \(communicating with us\)](#)
  - <https://piazza.com/class/idvnpk83p25421>

# Recitations on Wednesdays

---

- They are mandatory!
- Section A Sanjay Chandrasekaran
  - Wednesdays 9:30am Wean 5202
- Section B Tim Conley
  - Wednesdays 10:30am Wean5202

# How is Java different from Python?

---

- Indentation is not required, but strongly recommended
- Statements end with a semi-colon ‘;’
- Functions are called methods
- Java needs to be compiled
  - .java → [compiler] → .class
  - .class → [Java Virtual Machine] → program output
- Java programs “spread over” multiple connected files called classes

# Displaying the value of a variable

---

```
int z = 100;
```

```
System.out.print(z);
```

```
System.out.println("The value of z is " + z);
```

- The `+` is creating a new string of characters by appending one string to another

# Boolean Operators

---

## ■ Comparison Operators

- <
- <=
- >
- >=
- ==
- !=

## ■ Binary Operators

- and      &&
- or        ||
- not      !

# IF statement

---

## ■ IF statement

- if
  - something or nothing happens
- if - else
  - one of two possible things happens
- if - else if - else if - else if - else
  - one of more than two possible things happens

# For loop

---

```
int x;  
for(x = 0; x <= 10; x++){  
    System.out.println(x);  
}
```

```
for(initialize; test; increment){  
    do stuff ...  
}
```

# While loop

---

```
int x = 0;
while(x <= 10){
    System.out.println(x);
    x = x + 1;    //equivalent to x++ or x += 1
}
```

```
initialize
while( test ){
    do stuff ...
    increment
}
```



# Do loop

---

```
int x = 0;  
do{  
    System.out.println(x);  
    x = x + 1;    //equivalent to x++ or x += 1  
} while(x <= 10);
```

```
initialize  
do{  
    do stuff ...  
    increment  
} while( test );
```

# Wrapper Classes

---

- These classes in Java are used to convert a String object (a collection of chars) into a numerical value.

```
String value = "3.14159";
```

```
double PI = Double.parseDouble(value);
```

- Most of the time we use the Scanner Class to handle input. Sometimes, we gather input as text and then convert it to a corresponding numerical value
- Check out Java API for Integer and Double Classes

# Homework

---

- install software in your own computer ASAP!
  - Java (read pdf doc)
  - Eclipse (read pdf doc)
- Homework #1 is due Tuesday 9/08 by 11:50pm
- Quiz #1 on Wednesday 9/09 during recitation