CS106A: Programming Methodologies
Chris Piech

- Childhood: Nairobi, Kenya
- High School: Kuala Lumpur, Malaysia
- Stanford University Ph.D. in Neural Networks
- Research lab on AI for Social Good

My parents are interesting folks

I originally concentrated in graphics and worked at Pixar

The problem I really want to solve is to make high quality more education accessible
11 years ago to this day, I was sitting in your seats
Head TA: Julia Daniel
Course mechanics
(this is a light version. Please read the handout for details).
Course Website

http://cs106a.stanford.edu
Prerequisite Test
Lectures and Sections

- Weekly 50-min section led by awesome section leaders (the backbone of the class!)
- Signups begin on Thursday at 5:00pm and close Sunday at 5:00pm
Office Hours

LaIR: 7pm until 11pm Sunday through Thursday (starting next Sunday)

Piech, CS106A, Stanford University
Functionality and style grades for the assignments use the following scale:

++  A submission so good it “makes you weep.”
+
++  Satisfies all requirements of the assignment.
+  Meets most requirements, but with some problems.
✓−  Has more serious problems.
−  Is even worse than that.
---  Better than nothing.
What we will ask you to do

- Assignments: 45%
- Midterm: 15%
- Final: 30%
- Section Participation: 10%

*Two free late days
Optional Contest

Piech, CS106A, Stanford University
Textbook

*The Art & Science of Java* by Eric Roberts
- written here at Stanford
- tailored to this course
- a valuable reference
- usable on open-book exams

Karel the Robot Learns Java
- First week of material
CS106A Units

Hours per week = Units × 3
Average about 10 hours / week for assignments

Start Here

Are you an undergrad or SCPD student?

No

Do you want to take CS106A for fewer units?

Yes

3 Units - or -
4 Units

No

Yes

5 Units
Are you in the right place?
What is CS106A?
Computer science is no more about computers than astronomy is about telescopes, biology is about microscopes or chemistry is about beakers and test tubes. Science is not about tools, it is about how we use them and what we find out when we do.

- Michael Fellows and Ian Parberry
Learning Goals

- Learn how to harness computing power to solve problems.

To that end:

- Explore fundamental techniques in computer programming.
- Develop good software engineering style.
- Gain familiarity with the Java programming language.
There are a lot of cool programs you may one day write
Pat Hanrahan, one of the founders of Pixar is a professor here.
Consumer Applications
Autonomous Surgery
Self Driving Car
If only we could program self driving cars...
Graphical Games
Data Visualization

Piech, CS106A, Stanford University
Internet Based Program

Status: Chris is lecturing
Art of Computer Science
Strive for Everyone to Succeed
Meet Karel the Robot

Good morning
Karel Speaks Java

Piech, CS106A, Stanford University
Why Java?

1. Job postings containing top languages
   - Java: 68K
   - Python: 46K
   - JavaScript: 38K
   - C++: 33K

   Indeed.com - November, 17th 2017

2. [Link to Coding Dojo blog post](http://www.codingdojo.com/blog/7-most-in-demand-programming-languages-of-2018/)
Karel’s World

North

West

South

East
Walls

1 2 3 4

1 + + + +

2 + + + +

3 + + + +

Piech, CS106A, Stanford University
Beepers

1
2
3
1
2
3
4

Piech, CS106A, Stanford University
Knows Four Commands

move();
turnLeft();
putBeeper();
pickBeeper();
move();
move();
move();

```
+ + + +
+ + + +
+ + + +
+ + + +
+ + + +
```

1 2 3 4

1 2 3 4
turnLeft();
turnLeft();
'turnLeft();'
pickBeeper();
`pickBeeper();`
pickBeeper();
Make Sense?
First Challenge
First Challenge
Bird’s Eye View

1 2 3 4 5

1 2 3 4 5

1 2 3 4 5
Bird’s Eye View

Karel is facing East
Turn Left

Karel is facing North
Turn Left

Karel is facing West
Turn Left

Karel is facing South
First Challenge
First Challenge
First Challenge
Learn By Doing
private void name() {
    method statements
}

This adds a new command to Karels vocabulary
Anatomy of a Program

Import Packages

```java
public class OurKarelProgram extends Karel {
}
```
Anatomy of a Program

public class OurKarelProgram extends Karel {

    run method

    helper methods

}
public class OurKarelProgram extends Karel {

    public void run() {
        move();
        pickBeeper();
        move();
        turnLeft();
        move();
        turnRight();
        move();
        putBeeper();
        move();
    }

    // helper methods
}

Anatomy of a Program
Anatomy of a Program

Import Packages

```java
public class OurKarelProgram extends Karel {

    public void run() {
        move();
pickBeeper();
        move();
turnLeft();
        move();
turnRight();
        move();
putBeeper();
        move();
    }

    private void turnRight() {
        turnLeft();
turnLeft();
turnLeft();
turnLeft();
    }
}
```
import stanford.karel.*;

public class OurKarelProgram extends Karel {

    public void run() {
        move();
        pickBeeper();
        move();
        turnLeft();
        move();
        turnRight();
        move();
        putBeeper();
        move();
    }

    private void turnRight() {
        turnLeft();
        turnLeft();
        turnLeft();
    }

}
import stanford.karel.*;

public class OurKarelProgram extends Karel {

    public void run() {
        move();
        pickBeeper();
        move();
        turnLeft();
        move();
        turnRight();
        move();
        putBeeper();
        move();
    }

    private void turnRight() {
        turnLeft();
        turnLeft();
        turnLeft();
    }
}

This piece of the program's source code is called a method.
import stanford.karel.*;

public class OurKarelProgram extends Karel {

    public void run() {
        move();
        pickBeeper();
        move();
        turnLeft();
        move();
        turnRight();
        move();
        putBeeper();
        move();
    }

    private void turnRight() {
        turnLeft();
        turnLeft();
        turnLeft();
        turnLeft();
    }
}

This line of code gives the name of the method (here, run)
import stanford.karel.*;

public class OurKarelProgram extends Karel {

    public void run() {
        move();
        pickBeeper();
        move();
        turnLeft();
        move();
        turnRight();
        move();
        putBeeper();
        move();
    }

    private void turnRight() {
        turnLeft();
        turnLeft();
        turnLeft();
        turnLeft();
    }
}
import stanford.karel.*;

public class OurKarelProgram extends Karel {

    public void run() {
        move();
pickBeeper();
move();
turnLeft();
move();
turnRight();
move();
putBeeper();
move();
    }

    private void turnRight() {
        turnLeft();
turnLeft();
turnLeft();
turnLeft();
    }

}
import stanford.karel.*;

public class OurKarelProgram extends Karel {

    public void run() {
        move();
        pickBeeper();
        move();
        turnLeft();
        move();
        turnRight();
        move();
        putBeeper();
        move();
    }

    private void turnRight() {
        turnLeft();
        turnLeft();
        turnLeft();
    }
}
import stanford.karel.*;

public class OurKarelProgram extends Karel {

    public void run() {
        move();
        pickBeeper();
        move();
        turnLeft();
        move();
        turnRight();
        move();
        putBeeper();
        move();
    }

    private void turnRight() {
        turnLeft();
        turnLeft();
        turnLeft();
    }

}
import stanford.karel.*;

public class OurKarelProgram extends Karel {

    public void run() {
        move();
        pickBeeper();
        move();
        turnLeft();
        move();
        turnRight();
        move();
        putBeeper();
        move();
    }

    private void turnRight() {
        turnLeft();
        turnLeft();
        turnLeft();
    }

}
import stanford.karel.*;

public class OurKarelProgram extends Karel {

    public void run() {
        move();
        pickBeeper();
        move();
        turnLeft();
        move();
        turnRight();
        move();
        putBeeper();
        move();
    }

    private void turnRight() {
        turnLeft();
        turnLeft();
        turnLeft();
        turnLeft();
    }

}
Anatomy of a Program

```java
import stanford.karel.*;

public class OurKarelProgram extends Karel {

    public void run() {
        move();
pickBeeper();
        move();
turnLeft();
        move();
turnRight();
        move();
        putBeeper();
        move();
    }

    private void turnRight() {
        turnLeft();
        turnLeft();
        turnLeft();
        turnLeft();
    }
}
```

The `turnRight` method is “private” to indicate it is only visible to our current program.
Why Study CS?
Joy of Building
Interdisciplinary

Piech, CS106A, Stanford University
Closest Thing To Magic
Now is the Time
A machine learning algorithm performs **better than** the best dermatologists.

Developed this year, at Stanford.

Oh and Its Useful

1,000,000 more jobs than students by 2020

$500 billion opportunity

1.4 million computing jobs

400,000 computer science students

Code.org
Everyone is Welcome
The End
The End?