

CS106A: Programming Methodologies

Chris Piech

My parents are
interesting folks

I originally concentrated
in graphics and worked
at Pixar

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

The problem I really want to
solve is to make high quality
more education accessible



Chris Piech



11 years ago to this day, I was sitting in your seats

Piech, CS106A, Stanford University



Head TA: Julia Daniel



Piech, CS106A, Stanford University



Section Leaders



Piech, CS106A, Stanford University



Course mechanics

(this is a light version. Please read the handout
for details).

Course Website



<http://cs106a.stanford.edu>



Piech, CS106A, Stanford University



Prerequisite Test



Piech, CS106A, Stanford University

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



Piech, CS106A, Stanford University



Office Hours



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

Piech, CS106A, Stanford University



Grading Scale

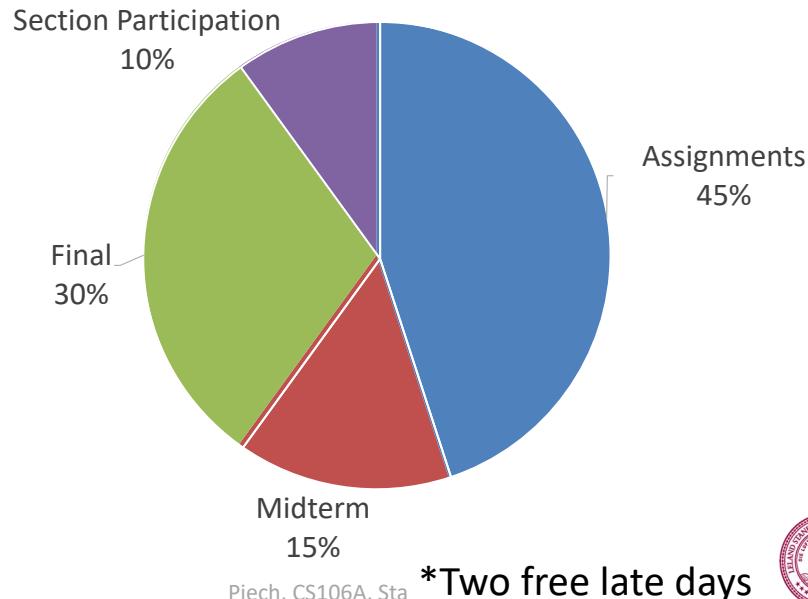
Functionality and style grades for the assignments use the following scale:

- ++** A submission so good it “makes you weep.”
- +** Exceeds requirements.
- ✓+** 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.

Piech, CS106A, Stanford University



What we will ask you to do



Piech, CS106A, Sta

Optional Contest



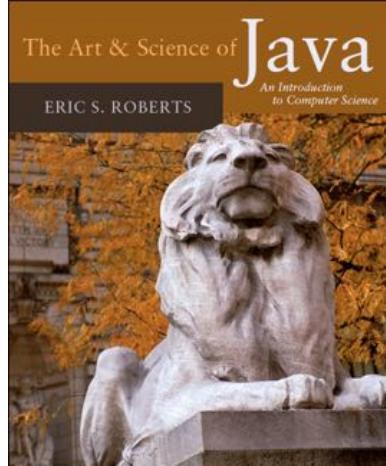
Piech, CS106A, Stanford University



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

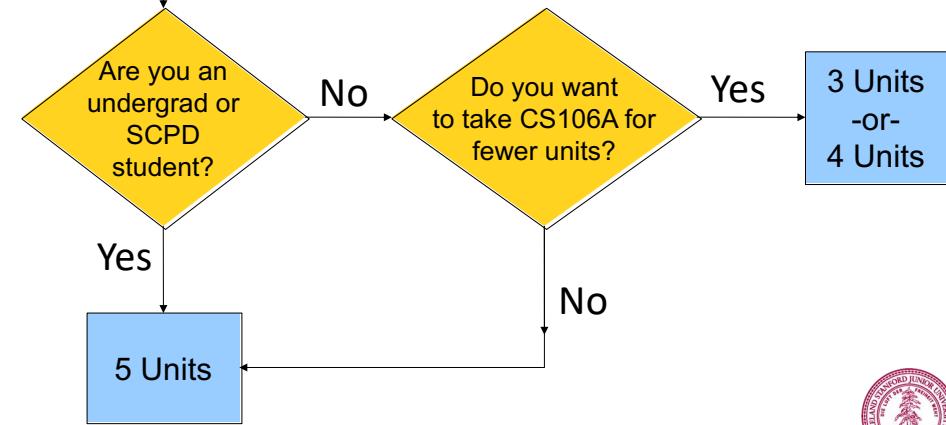
- First week of material

Piech, CS106A, Stanford University

CS106A Units

Hours per week = Units \times 3

Average about 10 hours / week for assignments



Piech, CS106A, Stanford University



Are you in the right place?

What is CS106A?

Computer Science

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.



Piech, CS106A, Stanford University

There are a lot of cool
programs you may one day
write

Computer Graphics



Pat Hanrahan, one of the founders of Pixar is a professor here



Piech, CS106A, Stanford University

Consumer Applications



Autonomous Surgery



Piech, CS106A, Stanford University

(c) 2012 Intuitive Surgical, Inc.



Self Driving Car

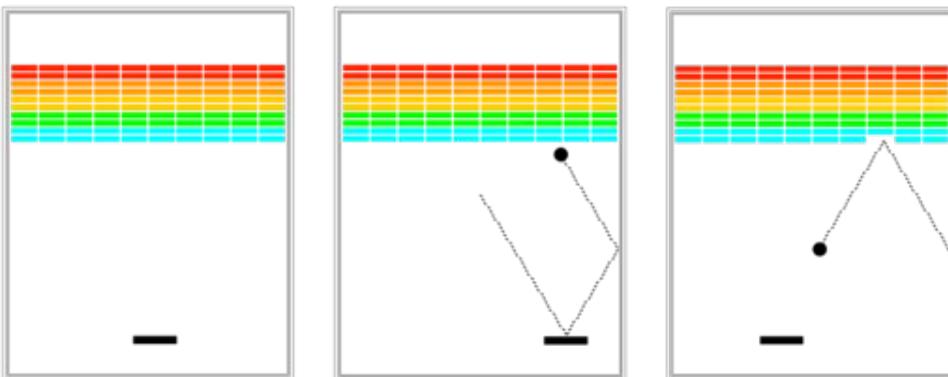


Piech, CS106A, Stanford University

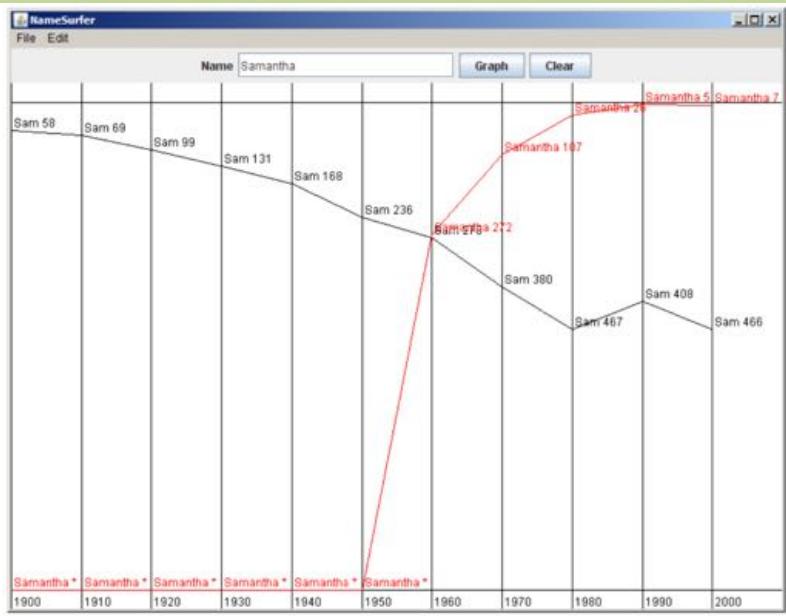


If only we could program self
driving cars...

Graphical Games



Data Visualization



Piech, CS106A, Stanford University



Internet Based Program



Piech, CS106A, Stanford University

Art of Computer Science



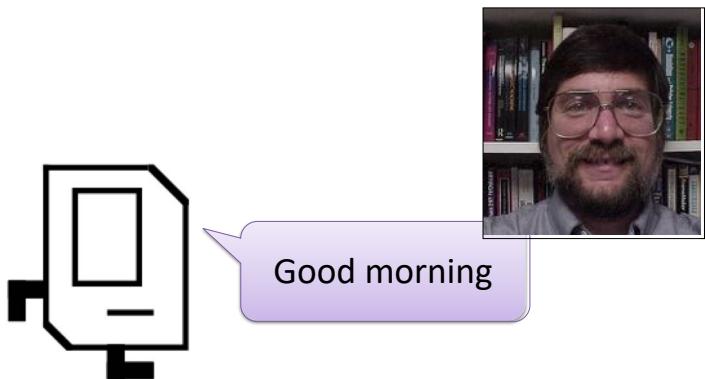
Strive for Everyone to Succeed

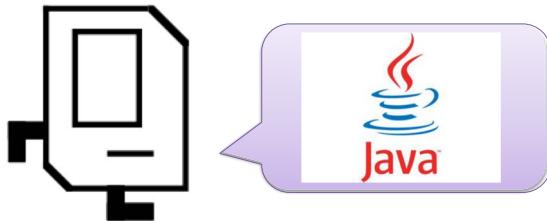


Lets Get Started



Meet Karel the Robot





Piech, CS106A, Stanford University

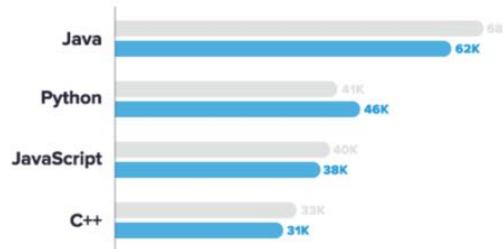


Why Java?

Job postings containing top languages

Indeed.com - November, 17th 2017

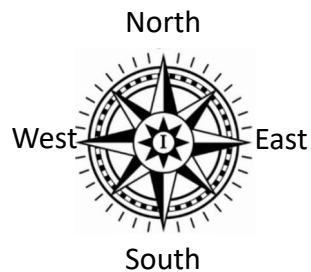
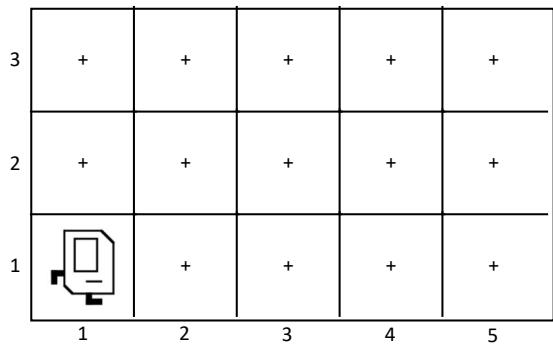
1



2



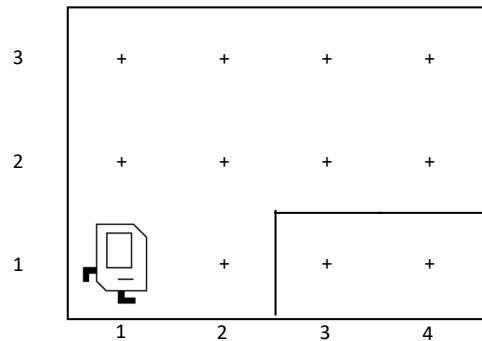
Karel's World



Piech, CS106A, Stanford University

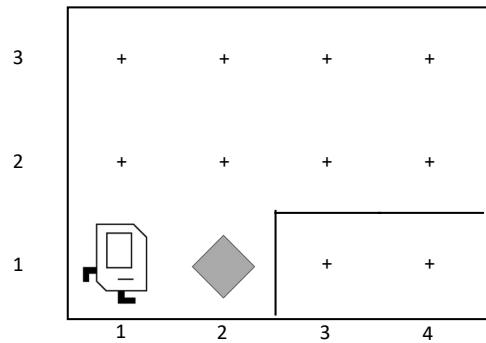


Walls



Piech, CS106A, Stanford University

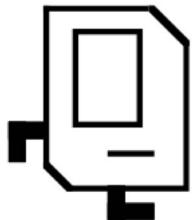
Beepers



Piech, CS106A, Stanford University



Knows Four Commands



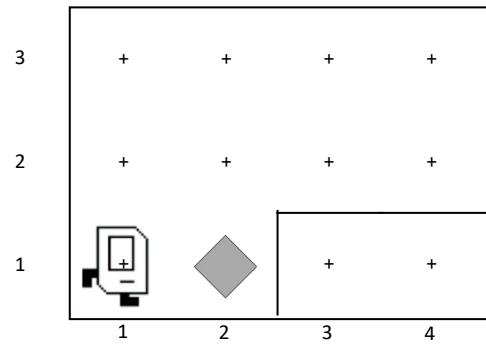
```
move();  
turnLeft();  
putBeeper();  
pickBeeper();
```

Piech, CS106A, Stanford University

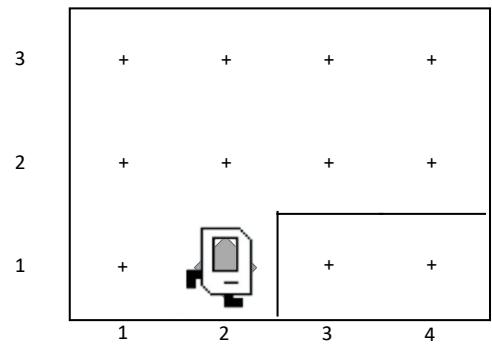


move();

move();



move();

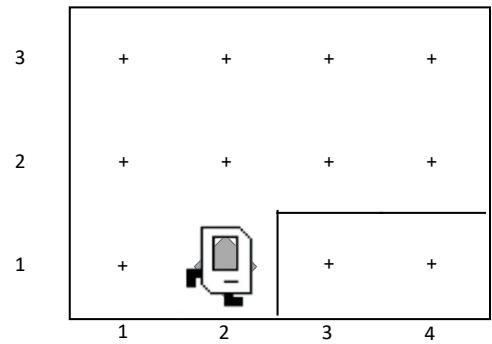


Piech, CS106A, Stanford University



turnLeft();

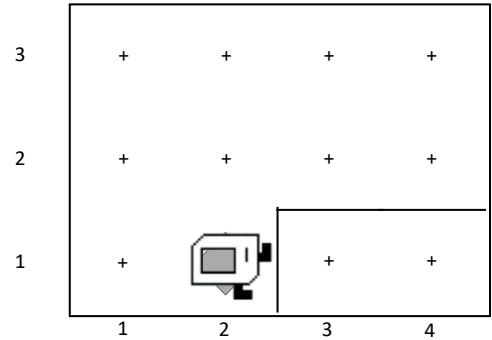
turnLeft();



Piech, CS106A, Stanford University



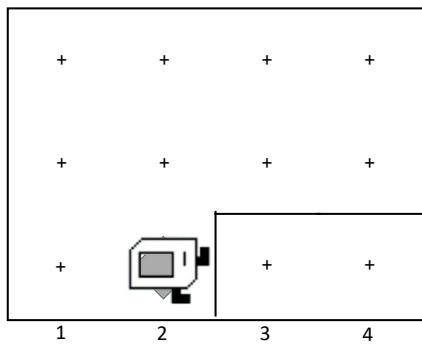
turnLeft();



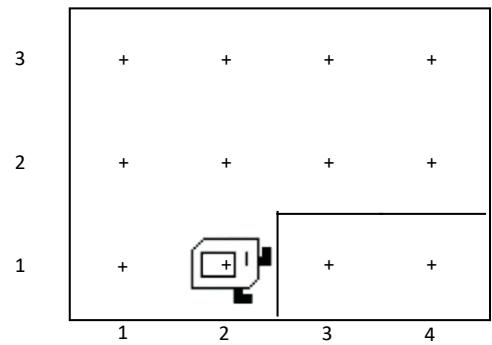
Piech, CS106A, Stanford University



```
pickBeeper();
```



pickBeeper();

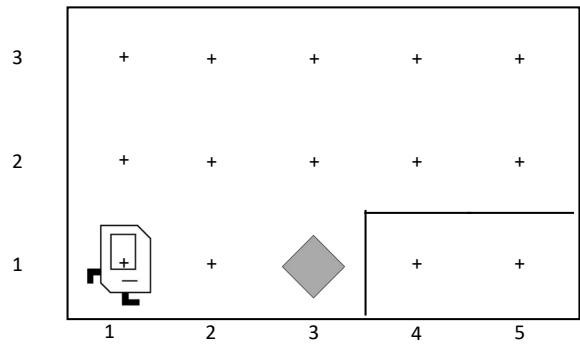


Piech, CS106A, Stanford University



Make Sense?

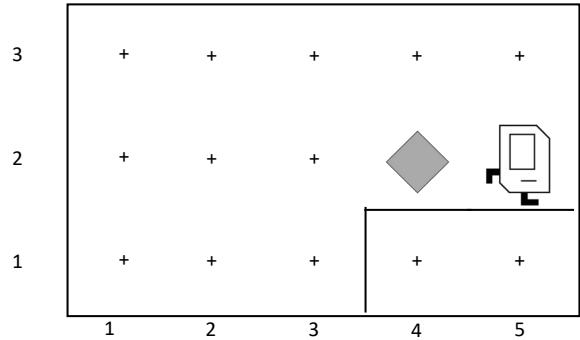
First Challenge



Piech, CS106A, Stanford University



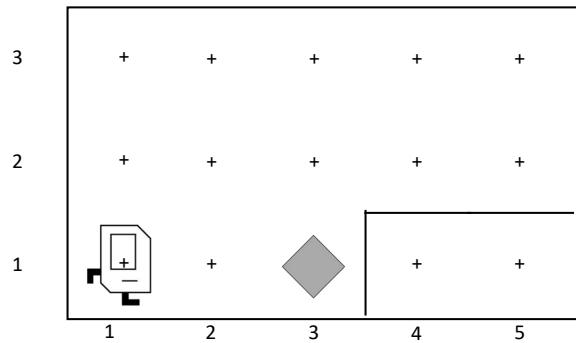
First Challenge



Piech, CS106A, Stanford University



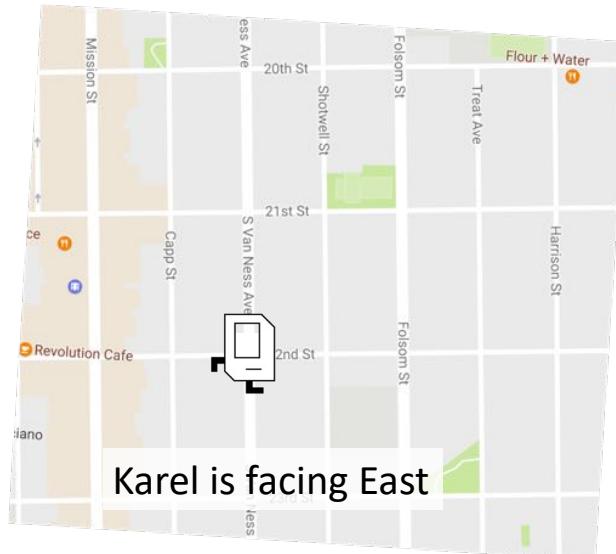
Bird's Eye View



Piech, CS106A, Stanford University



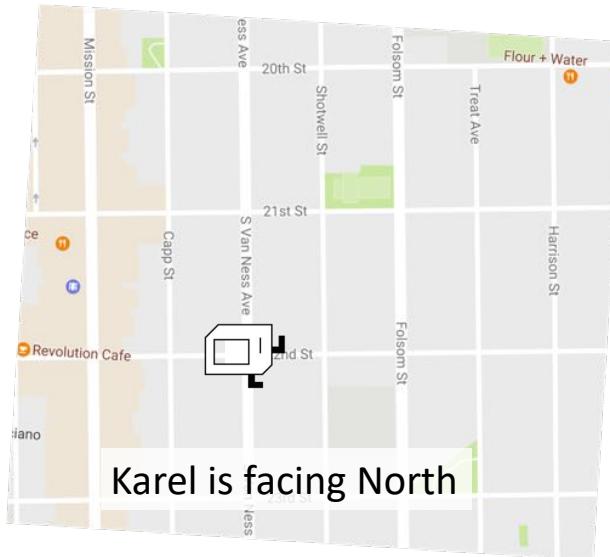
Bird's Eye View



Piech, CS106A, Stanford University



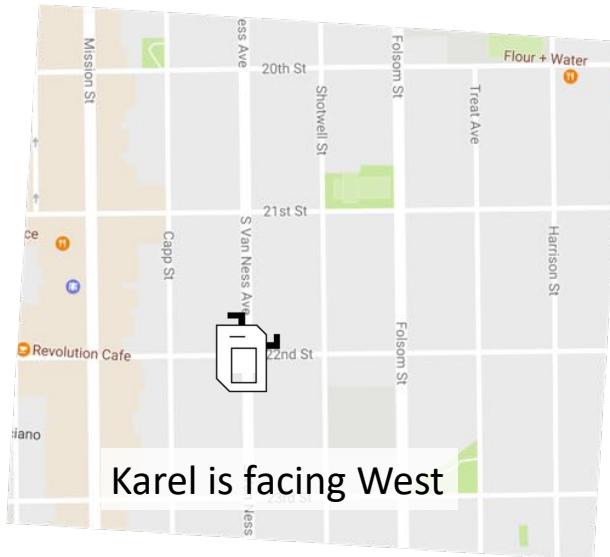
Turn Left



Piech, CS106A, Stanford University



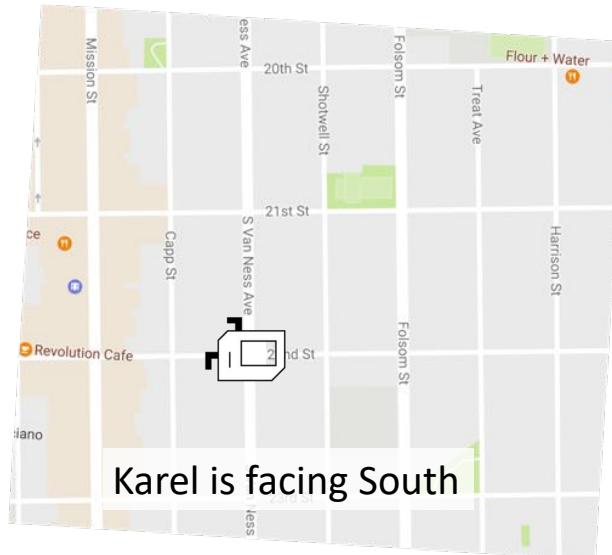
Turn Left



Piech, CS106A, Stanford University



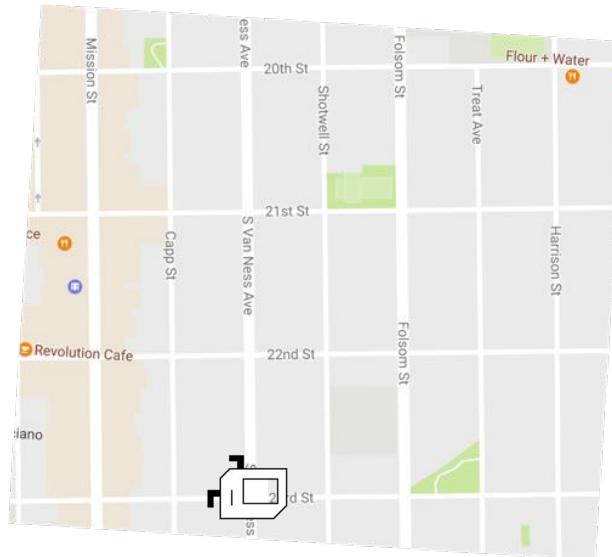
Turn Left



Piech, CS106A, Stanford University



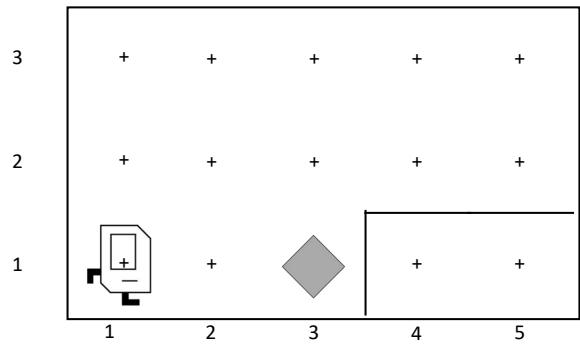
Move



Piech, CS106A, Stanford University



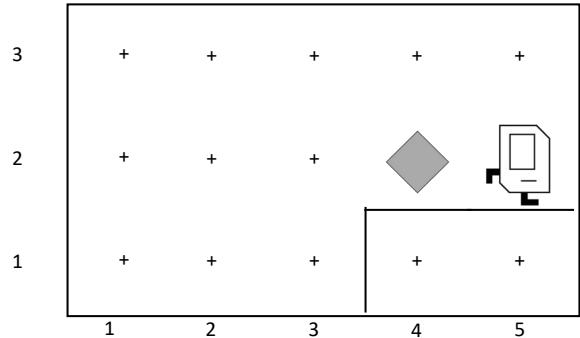
First Challenge



Piech, CS106A, Stanford University



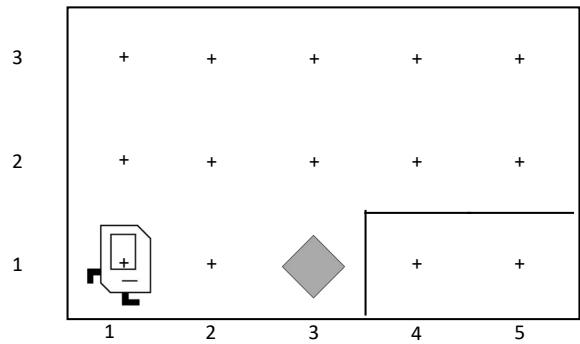
First Challenge



Piech, CS106A, Stanford University



First Challenge



Piech, CS106A, Stanford University



Learn By Doing





Piech, CS106A, Stanford University



Method Definition

```
private void name() {  
    method statements  
}
```

This adds a new
command to Karel's
vocabulary

Piech, CS106A, Stanford University



Anatomy of a Program

Import Packages

Program



Anatomy of a Program

Import Packages

```
public class OurKarelProgram extends Karel {
```

```
}
```



Anatomy of a Program

Import Packages

```
public class OurKarelProgram extends Karel {
```

run method

helper methods

}



Anatomy of a Program

Import Packages

```
public class OurKarelProgram extends Karel {
```

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

helper methods

}



Anatomy of a Program

Import Packages

```
public class OurKarelProgram extends Karel {  
  
    public void run() {  
        move();  
        pickBeeper();  
        move();  
        turnLeft();  
        move();  
        turnRight();  
        move();  
        putBeeper();  
        move();  
    }  
  
    private void turnRight() {  
        turnLeft();  
        turnLeft();  
        turnLeft();  
    }  
}
```



Anatomy of a Program

```
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();  
    }  
}
```



Anatomy of a Program

```
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**.



Anatomy of a Program

```
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 line of code gives the **name** of the method
(here, run)



Anatomy of a Program

```
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 line of code gives the
name of the method
(here, turnRight)



Anatomy of a Program

```
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 is called a **code
block**



Anatomy of a Program

```
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 is called a *code block*



Anatomy of a Program

```
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 is also called a
code block



Anatomy of a Program

```
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();  
    }  
}
```



Anatomy of a Program

```
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();  
    }  
}
```

The run method is “public” so that
Eclipse can call it.



Anatomy of a Program

```
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();  
    }  
}
```

The turnRight method is “private” to indicate it is only visible to our current program.



Why Study CS?

Joy of Building



Interdisciplinary



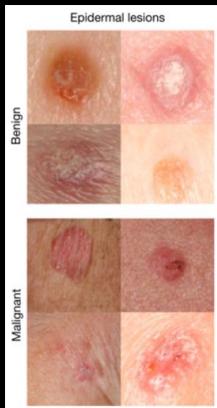
Closest Thing To Magic



Now is the Time



Now is the Time

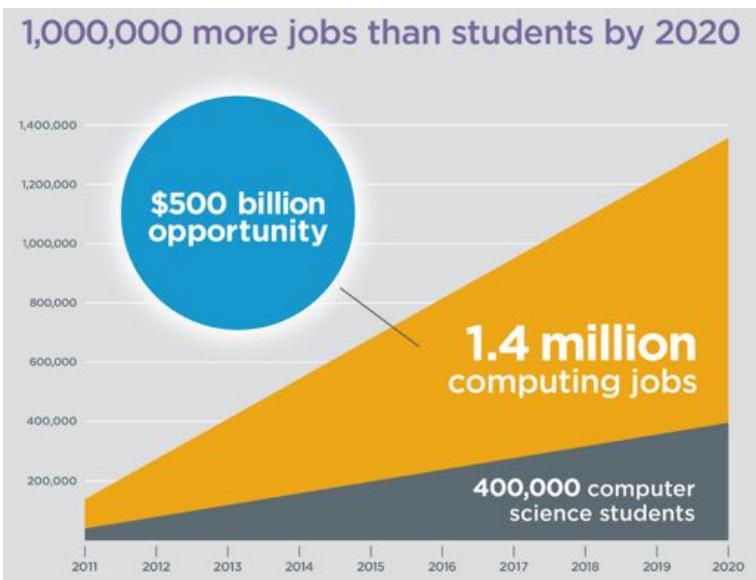


A machine learning algorithm performs **better than** the best dermatologists.

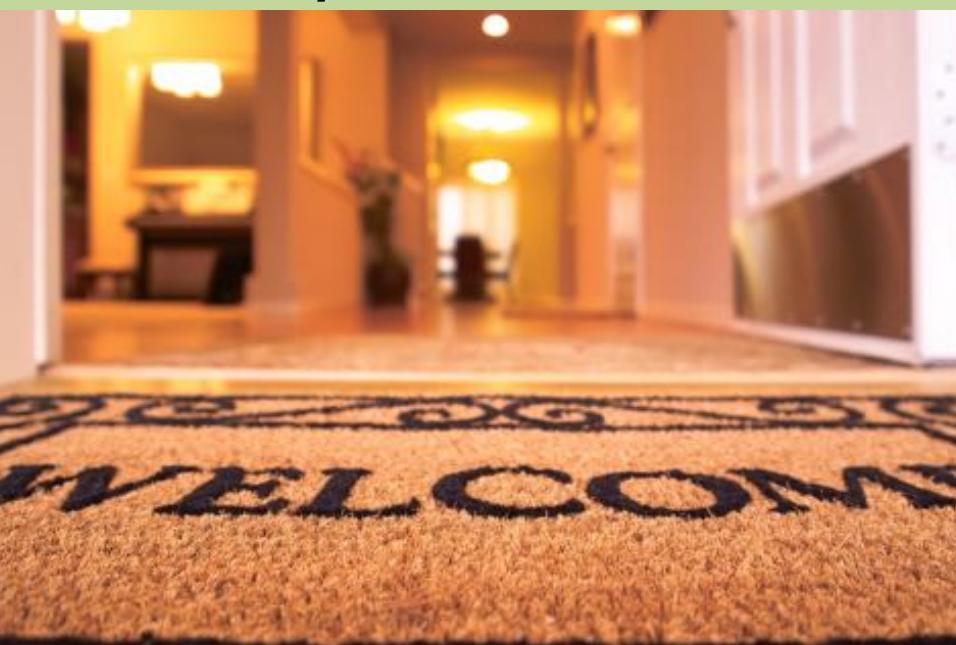
Developed this year, at Stanford.

Esteva, Andre, et al. "Dermatologist-level classification of skin cancer with deep neural networks." *Nature* 542.7639 (2017): 115-118.

Oh and Its Useful



Everyone is Welcome



The End

A close-up photograph of a silver-colored spinning top with a dark base, captured in mid-spin against a dark, blurred background. The text "The End?" is overlaid in white.

The End?