

Introduction to CS 106J

Eric Roberts and Jerry Cain
CS 106J
April 3, 2017

Course Description

CS 106J: Programming Methodology in JavaScript

Introduction to the engineering of computer applications emphasizing modern software engineering principles: object-oriented design, decomposition, encapsulation, abstraction, and testing. Emphasis is on good programming style. This course covers the same material as CS 106A but does so using JavaScript, the most common language for implementing interactive web pages, instead of Java. No prior programming experience required. Enrollment limited to 100.

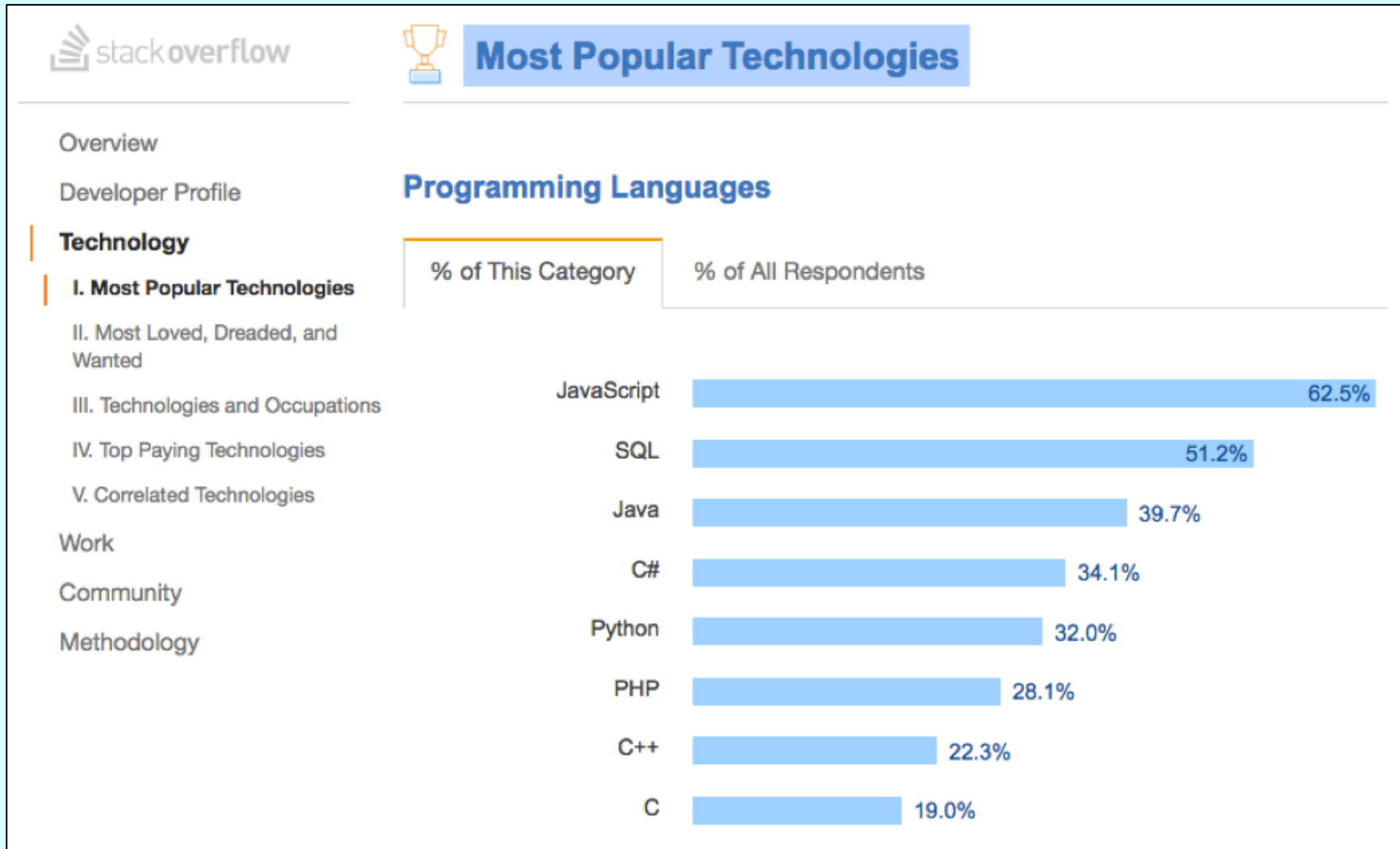
Terms: Spr | Units: 3-5 | UG Reqs: WAY-FR | Grading: Letter or CR/NC

CS 106J covers the same material and meets the same WAYS requirements as any other CS 106A section.

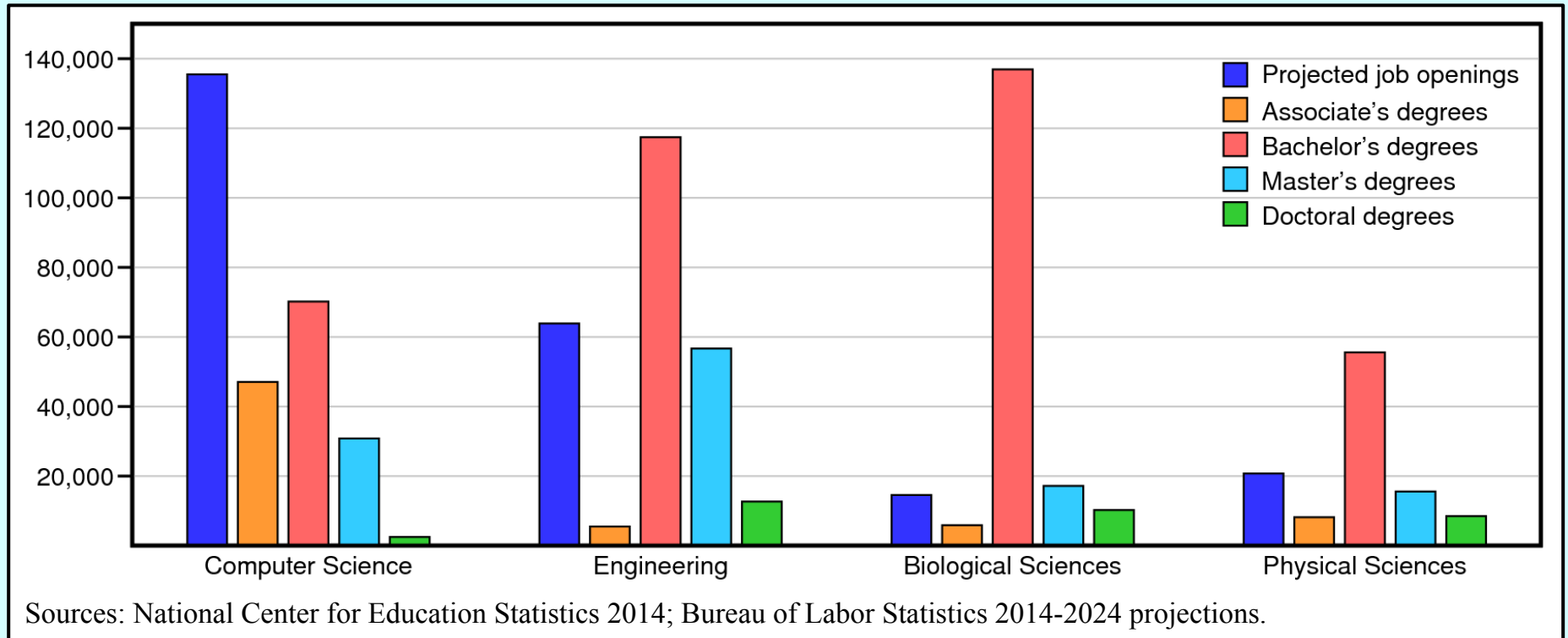
Why JavaScript?

- When Stanford adopted Java a little over a decade ago, we expected—along with its designers—that it would become the “language of the web.” That didn’t happen.
- Today, the “language of the web” is JavaScript, which has become the most widely used language in industry.
- Along with JavaScript expert Douglas Crockford, we believe that, as long as you avoid some of its most commonly abused features, JavaScript is “a beautiful, elegant, highly expressive language” that is ideal for a first course in programming.
 - It is considerably easier to learn than Java.
 - There are far fewer confusing details to memorize.
 - It offers much cleaner implementations of modern features.
 - It is universally supported on the web.

JavaScript is the Most Popular Language



Why Study Computer Science?

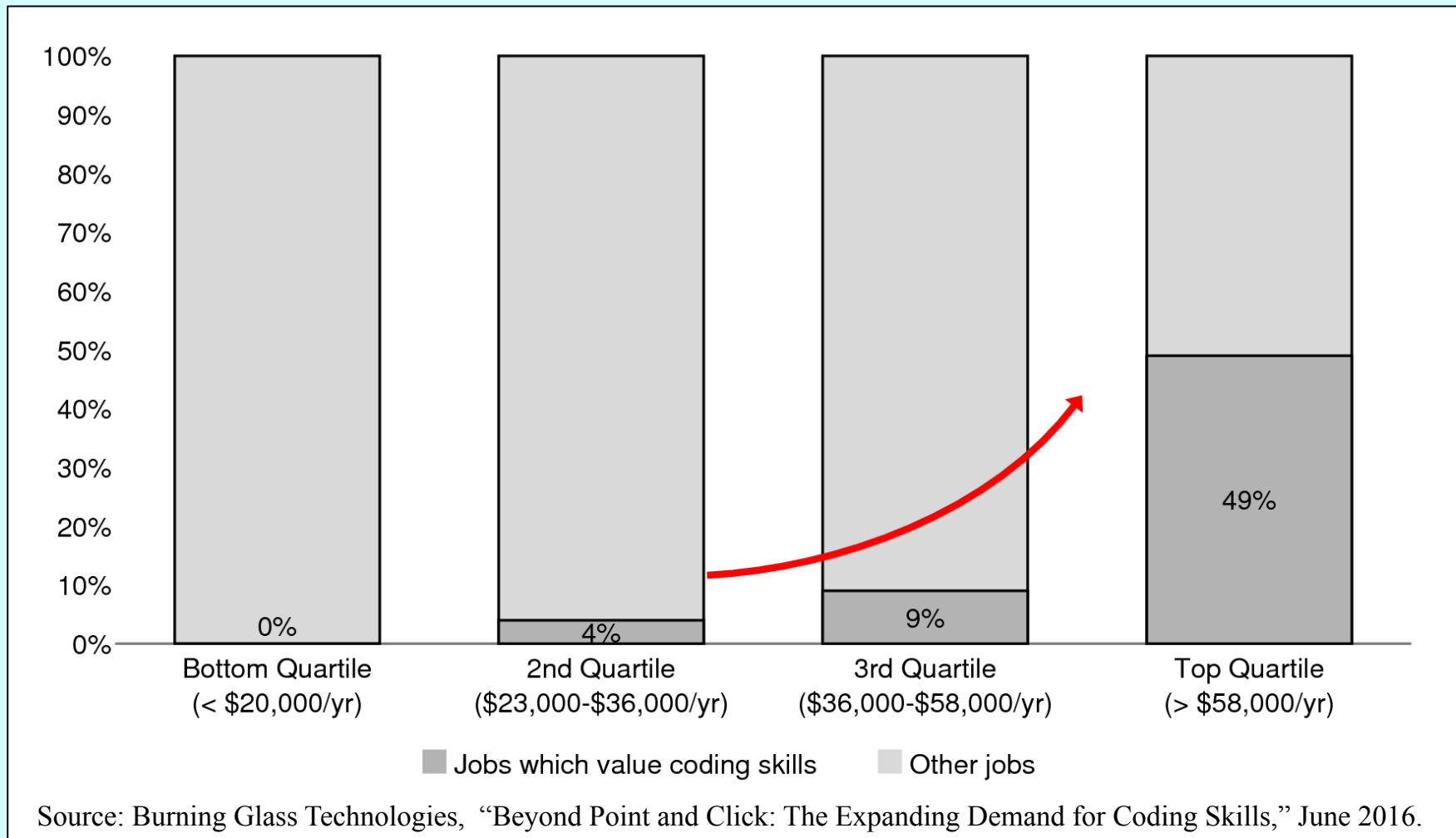


We are very happy with the students that we get from this university. . . . We just wish we could hire two to three times as many of them.

— Bill Gates at Stanford, February 19, 2008

Everyone Needs Some Programming

Half of all jobs in the top income quartile value coding skills.



CS 106J Course Staff



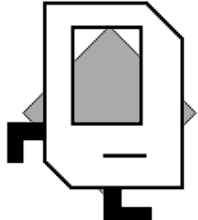
Eric Roberts
eroberts@cs.stanford.edu
Office Hours (Gates 202):
Tuesdays 9:30–11:00



Jerry Cain
jerry@cs.stanford.edu
Office Hours (Gates 192):
Wednesdays: 3:30–5:00

Jason Chen
cheson@stanford.edu
Office Hours (TBA):
TBA

Syllabus—Week 1

<p>April 3</p> <p>Administration Course overview Meet Karel the Robot</p> 	<p>5</p> <p>Simple Karel programs Control structures in Karel</p> <p>Read: Sections 1.1-1.3</p>	<p>7</p> <p>Problem-solving in Karel Program decomposition The idea of an algorithm</p> <p>Read: Sections 1.4-1.5</p>
--	--	--

Syllabus—Week 2

<p>10</p> <p>Programming by example Variables and values Arithmetic expressions Functions</p> <p>Read: Sections 2.1-2.4</p>	<p>12</p> <p>Strings and concatenation Simple graphics</p> <p>Read: Sections 2.5-2.6</p>	<p>14</p> <p>Control statements Boolean data</p> <p>Read: Sections 3,1-3.5</p>
--	---	---

Syllabus—Week 3

<p>17</p> <p>Arguments and parameters Libraries Random numbers</p> <p>Read: Sections 4.1-4.2 Due: HW #1 (Karel)</p>	<p>19</p> <p>The mechanics of functions Decomposition</p> <p>Read: Sections 4.3-4.4 Due: Karel contest</p>	<p>21</p> <p>First-class functions Event-driven programming Responding to mouse events</p> <p>Read: Sections 5.1-5.4</p>
---	--	---

Syllabus—Week 4

24

Simple animation
Timers

Read: Section 5.5

Due: HW #2 (Simple JS)

26

The **GArc** class
The **GPolygon** class
The **GCompound** class

Read: Section 5.6

28

Binary representation
Representing characters

Read: Section 6.1

Syllabus—Week 5

<p>May 1</p> <p>JavaScript's String class Common string patterns</p> <p>Read: Sections 6.2-6.3</p>	<p>3</p> <p>Problem solving with strings String applications</p> <p>Read: Section 6.4</p>	<p>5</p> <p>Cryptography The GLabel class</p> <p>Read: Section 6.5 Due: HW #3 (Breakout)</p>
--	--	---

Syllabus—Week 6

8	10	12
Debugging strategies	Simple arrays and files	Using arrays for tabulation
Midterm Exam Tuesday, May 9 3:30 P.M. or 7:00 P.M.		
Due: Graphics contest	Read: Sections 7.1-7.2	Read: Section 7.3

Syllabus—Week 7

<p>15</p> <p>Multidimensional arrays Pixel arrays The GImage class</p> <p>Read: Sections 7.4-7.5 Due: HW #4 (HangKarel)</p>	<p>17</p> <p>Objects as aggregates</p> <p>Read: Sections 8.1-8.2</p>	<p>19</p> <p>JavaScript and OOP Libraries and interfaces Defining classes and methods</p> <p>Read: Sections 8.3-8.5</p>
--	---	--

Syllabus—Week 8

<p>22</p> <p>Objects as maps</p> <p>Read: Section 8.6</p>	<p>24</p> <p>Large-scale data structures Data-driven programs</p> <p>Read: Section 8.7</p>	<p>26</p> <p>Overview of Adventure!</p> <p>Read: Adventure handout Due: HW #5 (Yahtzee)</p>
--	---	---

Syllabus—Week 9

<p>29</p> <p>Memorial Day</p> <p>(no class)</p>	<p>31</p> <p>JavaScript and the web</p> <p>Read: (handouts for the day)</p>	<p>June 2</p> <p>Using web interactors</p> <p>Read: (handouts for the day)</p>
---	--	---

Dead Week and Beyond

<p>5</p> <p>Looking ahead</p> <p>Read: (handouts for the day)</p>	<p>7</p> <p>Frontiers of computing (optional)</p> <p>Due: HW #6 (Adventure) Due: Adventure contest</p>	
--	--	--

Review session:
Monday, June 12
7:00-9:00 P.M.

Final exam:
Wednesday, June 14
8:30-11:30 A.M.

Assignments in CS 106J

- Assignments in CS 106J are due at 5:00P.M. Assignments that come in after 5:00 will be considered late.
- Everyone in CS 106J starts the quarter with two “late days” that you can use at any time you need some extra time. In my courses, late days correspond to class meetings, so that, if an assignment is due on Wednesday and you turn it in on Friday, that counts as *one* late day.
- Extensions can be approved only by the TA, Jason Chen.
- Assignments are graded by your section leader, who discusses your work in an interactive, one-on-one grading session.
- Each assignment is given two grades: one on functionality and one on programming style. Style matters. Companies in Silicon Valley expect Stanford graduates to understand how to write code that other programmers can maintain.

The CS 106J 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.
 - ✓-** Some more serious problems.
 - Even worse than that.
 - Why did you turn this in?

Contests

- CS 106J will have three contests as follows:
 - The Karel Contest associated with Assignment #1
 - The Graphics Contest associated with Assignment #3
 - The Adventure Contest associated with Assignment #6
- First prize in the contest is a score of 100% on one of the graded components of the course, typically the final exam.
- As an additional incentive, entering any of the contests gives you a virtual ticket to win an additional grand prize in a random drawing at the end of the quarter. So does receiving a runner-up or honorable mention on a contest and finding errors in the text and tools.
- Entering a contest also earns “house points” for your class in the style of the Hogwarts School from Harry Potter.

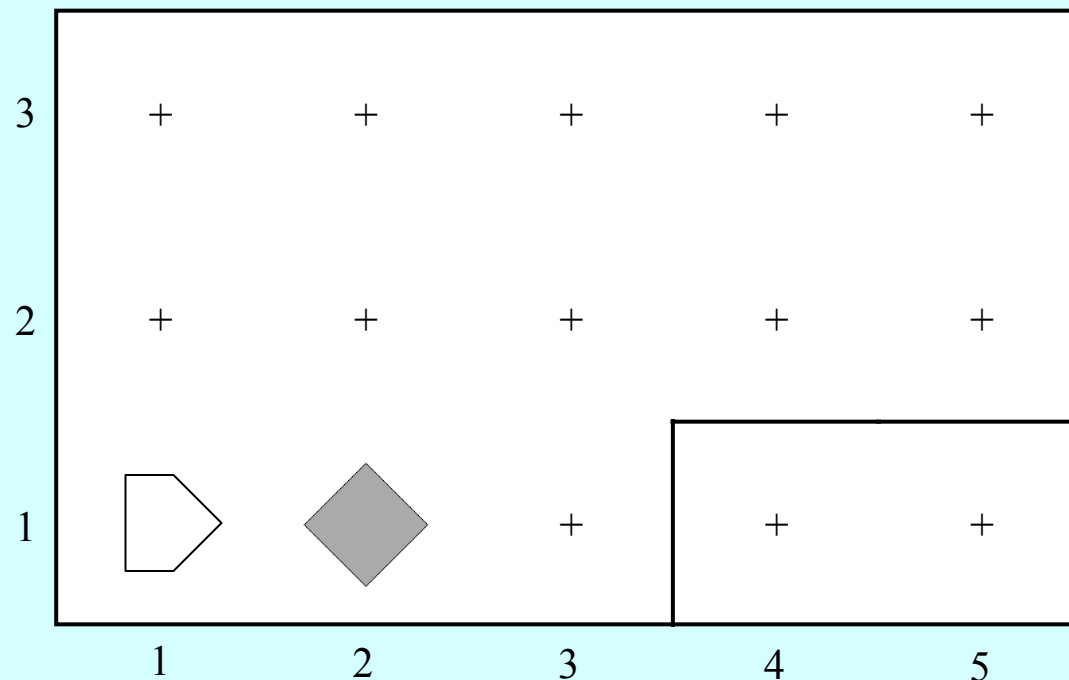
Honor Code Rules

- Rule 1: You must not look at solutions or program code that is not your own.
- Rule 2: You must not share your solution code with other students.
- Rule 3: You must indicate on your submission any assistance you received.

Meet Karel the Robot

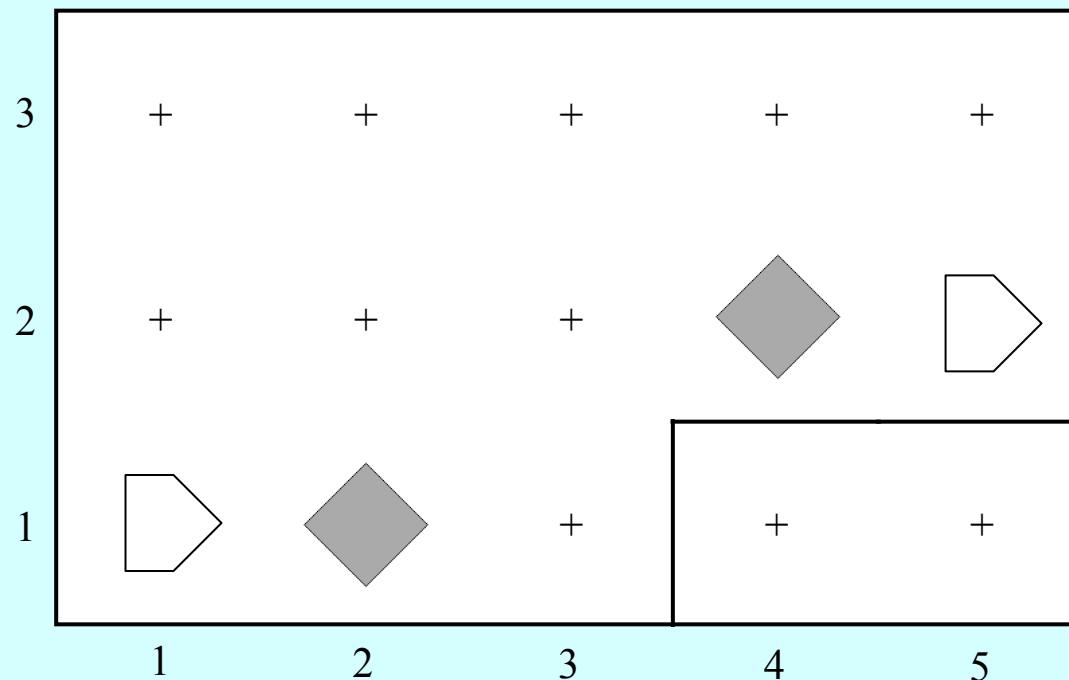
- Initially, Karel understands only four primitive commands:

<code>move ()</code>	Move forward one square
<code>turnLeft ()</code>	Turn 90 degrees to the left
<code>pickBeeper ()</code>	Pick up a beeper from the current square
<code>putBeeper ()</code>	Put down a beeper on the current square



Your First Challenge

- How would you program Karel to pick up the beeper and transport it to the top of the ledge? Karel should drop the beeper at the corner of 2nd Street and 4th Avenue and then continue one more corner to the east, ending up on 5th Avenue.



The End