

# RYAN LEE

Phone: 510-780-6526  
Website: ryanlee.github.io  
Email: ryanlee@berkeley.edu

## EDUCATION

---

**University of California, Berkeley**  
Double Major: Computer Science, Cognitive Science  
GPA: 3.2/4.0

GRADUATING 2018

**Alameda High School**  
Class rank: 1 out of 431  
GPA: 4.85/5.0 (weighted), 4.0/4.0 (unweighted)

GRADUATED 2014

## COURSEWORKS (COMPLETED / In Progress)

---

(CS 61a) Structure & Interpretation of Computer Programs	(CS 61b) Data Structures
(CS 198) iOS App Development	(CS 70) Discrete Math & Probability Theory
(Math 54) Linear Algebra & Differential Equations	(CS 98) Web Design
(CS 170) <i>Efficient Algorithms &amp; Intractable Problems</i>	(CS 188) <i>Artificial Intelligence</i>

## SKILLS

---

LANGUAGES	<b>Computer Languages</b> Fluent: Java, Python, Objective C, HTML/CSS Proficient: Javascript, Scheme, SQL, $\text{\LaTeX}$	<b>Spoken Languages</b> Cantonese, Spanish, English
-----------	--	--

## PROJECTS

---

SPRING 2013 (PERSONAL)	<b>Jellyfishing Game</b> <ul style="list-style-type: none"><li>• Built a catching game with the option for endless play or levels</li><li>• Self taught basic python and pygame to create this keyboard based game</li><li>• Player uses the arrow keys to move a net around the screen and catch floating jellyfish</li><li>• As levels progress, jellyfish move faster, and jellyfish amount and time limit both increase</li></ul>
FALL 2015 (ACADEMIC)	<b>Assembler and CPU</b> <ul style="list-style-type: none"><li>• Wrote an assembler to translate MIPS instructions to machine code</li><li>• Formed linker to process object files and generate an executable file in the end</li><li>• Utilized Logisim to create the register files and ALU for the computer processor unit</li><li>• Laid down the datapath, control, and pipeline to finish the 32-bit two-cycle CPU</li></ul>
SPRING 2015 (ACADEMIC)	<b>Gitlet</b> <ul style="list-style-type: none"><li>• Coded together own version control system that mimics Git</li><li>• Devised an immutable commit tree to hold on to individual commit nodes</li><li>• Supports “init,” “add,” “commit,” “remove,” “log,” “find,” “status,” “checkout,” and “branch”</li></ul>
FALL 2014 (ACADEMIC)	<b>Scheme Interpreter</b> <ul style="list-style-type: none"><li>• Assembled an interpreter to read in a subset of the Scheme functional programming language</li><li>• Written in Python, parses through lines of Scheme expressions and evaluates them</li><li>• Returns the results of these expressions in the end</li></ul>

## AWARDS AND HONORS

---

2014 **Valedictorian, Alameda High School**  
2013 **NorCal DECA Exam Winner, DECA Club**  
2013 **Scholar Athlete, Alameda High School**