


# The Beauty and Joy of Computing

## Lecture #1 Welcome; Abstraction

**UC Berkeley EECS  
Sr Lecturer SOE  
Dan Garcia**

**BJC: YOU'LL LOVE IT!**

Watch the student testimonials about the course, what it means to them, and how it has changed their lives. Inspiring!



[inst.eecs.berkeley.edu/~cs10/](http://inst.eecs.berkeley.edu/~cs10/)

## BJC in one slide

- Big Ideas of Programming**
  - Abstraction
  - Algorithms (2)
  - Recursion (2)
  - Functions-as-data,  $\lambda$ , (2)
  - Programming Paradigms
  - Concurrency
  - Distributed Computing
- Big Ideas of Computing**
  - HowStuffWorks
    - 3D Graphics + Video Games
    - Internet
  - Research Summaries
    - AI
    - HCI
  - The Power of Data (big, small, etc)
  - Apps that Changed the World
  - Social Implications of Computing
  - Saving the World with Computing
  - Cloud Computing
  - Limits of Computing
  - Future of Computing

**Beauty and Joy**

- "CS Unplugged" activities
- All lab work in pairs
- Two 3-week projects in pairs
  - Of their own choice!! (data + prog)
- One writeup
  - Of students' own choice!!





UC Berkeley "The Beauty and Joy of Computing" : Welcome, Abstraction (2)

## Format & Textbooks

- Format (7 hrs/wk \* 14 wks)**

| Mon     | Tue | Wed     | Thu | Fri        |
|---------|-----|---------|-----|------------|
| Lecture | Lab | Lecture | Lab | Discussion |
|         | Lab |         | Lab |            |

- Selected Reading**
  - Taken from great book "Blown to Bits" by Abelson, Ledeen & Lewis + articles + videos
  - Current events EVERY LECTURE (e.g., IBM's Watson vs Jeopardy)
- All resources FREE**
  - Even clickers!
- Pair Programming!**

UC Berkeley "The Beauty and Joy of Computing" : Welcome, Abstraction (3)

## Week at a glance (cue calendar)

UC Berkeley "The Beauty and Joy of Computing" : Welcome, Abstraction (4)


## Pro-student Grading Policies

- EPA**
  - Rewards good behavior
  - Effort
    - E.g., Office hours, doing every single lab, hw, reading Piazza pages
  - Participation
    - E.g., Raising hand in lec or discussion, asking questions on Piazza
  - Altruism
    - E.g., helping other students in lab, answering questions on Piazza
- You have 3 "Slip Days"**
  - You use them to extend due date, 1 slip day for 1 day extension
  - You can use them one at a time or all at once or in any combination
  - They follow you around when you pair up (you are counted individually)
    - E.g., A has 2, B has 0. Project is late by 1 day. A uses 1, B is 1 day late
  - Late is 1/3 off/day

UC Berkeley "The Beauty and Joy of Computing" : Welcome, Abstraction (5)

## Abstraction


- Detail removal**
  - "The act or process of leaving out of consideration one or more properties of a complex object so as to attend to others."
- Generalization**
  - "The process of formulating general concepts by abstracting common properties of instances"



Henri Matisse "Naked Blue IV"

UC Berkeley "The Beauty and Joy of Computing" : Welcome, Abstraction (6)

## Detail Removal



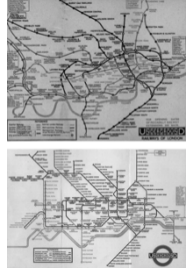
General Purpose Online Map      Selected Roads      Our Result

**Automatic Generation of Detail Maps**  
Maneesh Agrawala (UCB EECS), among others

UC Berkeley "The Beauty and Joy of Computing": Welcome, Abstraction (10)

## Detail Removal (in BJC)

- You'll want to write a project to simulate a real-world situation, or play a game, or ...
- Abstraction is the idea that you focus on the essence, the cleanest way to map the messy real world to one you can build
- Experts are often brought in to know what to remove and what to keep!

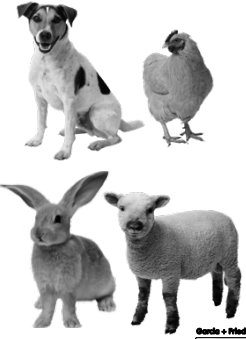


The London Underground 1928 Map & the 1933 map by Harry Beck.

UC Berkeley "The Beauty and Joy of Computing": Welcome, Abstraction (11)

## Generalization Example

- You have a farm with many animal kinds.
- Different food for each
- You have directions that say
  - To feed dog, put dog food in dog dish
  - To feed chicken, put chicken food in chicken dish
  - To feed rabbit, put rabbit food in rabbit dish
  - Etc...
- How could you do better?
  - To feed <animal>, put <animal> food in <animal> dish



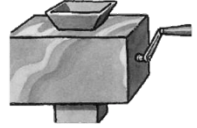
UC Berkeley "The Beauty and Joy of Computing": Welcome, Abstraction (12)

## Generalization (in BJC)

- You are going to learn to write functions, like in math class:

$$y = \sin(x)$$

- You should think about what inputs make sense to use so you don't have to duplicate code



"Function machine" from *Simply Scheme* (Harvey)

UC Berkeley "The Beauty and Joy of Computing": Welcome, Abstraction (13)

## The Power of Abstraction, everywhere!

- Examples:**
  - Functions (e.g.,  $\sin x$ )
  - Hiring contractors
  - Application Programming Interfaces (APIs)
  - Technology (e.g., cars)
- Amazing things are built when these layer**
  - And the abstraction layers are getting deeper by the day!

*We only need to worry about the interface, or specification, or contract NOT how (or by whom) it's built*

**Above the abstraction line**

**Abstraction Barrier (Interface)**  
(the interface, or specification, or contract)

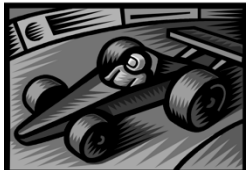
**Below the abstraction line**

*This is where / how / when / by whom it is actually built, which is done according to the interface, specification, or contract.*

UC Berkeley "The Beauty and Joy of Computing": Welcome, Abstraction (14)

## Summary

- Abstraction is one of the big ideas of computing and computational thinking
- Think about driving. How many of you know how a car works? How many can drive a car? Abstraction!



Someone who drove in 1930 could still drive a car today because they've kept the same Abstraction!  
(right pedal faster, left pedal slow)

UC Berkeley "The Beauty and Joy of Computing": Welcome, Abstraction (15)