



# The Beauty and Joy of Computing

Lecture #1  
Welcome; Abstraction


**BJC: YOU'LL LOVE IT! HARD WORK, BUT FUN!**  
Watch the student testimonials about the course, what it means to them, and how it has changed their lives. It's quite Inspiring!



Jon McKinsey



Michael Ball




UC Berkeley "The Beauty and Joy of Computing" : Welcome, Abstraction (1)

# Computing in the News: BJC4NYC

- We're working with the NYC school district to bring BJC to 100 NYC teachers by 2020!
- We were just mentioned by White House!

[bjc.link/bjcwhitehouse](http://bjc.link/bjcwhitehouse)




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

# (AP) Computer Science Principles

**7 Big Ideas**

- Creativity
- Abstraction
- Data and Information
- Algorithms
- Programming
- The Internet
- Global Impact




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

# (AP) Computer Science Principles

**6 Computational Thinking Practices**


- Connecting computing
- Creating computing artifacts
- Abstracting
- Analyzing problems & artifacts
- Communicating
- Collaborating



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

# Beauty and Joy of Computing in one slide

- Big Ideas of Programming**
  - Abstraction
  - Algorithms (2)
  - Recursion (2)
  - Functions-as-data, I (2)
  - Programming Paradigms
  - Cloud Computing
- Beauty and Joy**
  - "CS Unplugged" activities
  - If t2t, all lab work in pairs
  - "Create" performance task
    - Of your own choice!!
  - "Explore" performance task
    - Of your own choice!!
- Big Ideas of Computing**
  - Daily "computing in the news"
  - How the Internet works
  - Research Summaries
    - AI
    - HA
  - The Power of Data (big, small)
  - Social Implications of Computing
  - Saving the World w/Computing
  - Cloud Computing
  - Limits of Computing
  - Future of Computing
  - Robots...



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

# Incredibly easy-to-learn coding in Snap!



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

## Format & Textbooks

- **Format (14 hrs/wk \* 7 wks)**
  - See the following slides
- **Selected Reading**
  - Taken from book "Blown to Bits" by Abelson, Ledeen & Lewis + articles + videos
  - Current events EVERY LECTURE (e.g., IBM's Watson vs Jeopard)
- **All resources FREE**
  - Even dickers!
- **Pair Programming!**



UC Berkeley "The Beauty and Joy of Computing" : Welcome, Abstraction (7)

## Weekly Schedule (on cs10.org)

UC Berkeley "The Beauty and Joy of Computing" : Welcome, Abstraction (8)

## Let's check enrollments...

- We have NEVER turned anyone away, and we don't intend to turn anyone away this semester!

UC Berkeley "The Beauty and Joy of Computing" : Welcome, Abstraction (9)

## Peer Instruction for active learning!

- Increase real-time learning in lecture, test understanding of concepts vs. details
- As complete a "segment" ask multiple choice question
  - 1-2 minutes to decide yourself
  - 2 minutes in pairs/triples to reach consensus. Teach others!
  - 2 minute discussion of answers, questions, clarifications



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

## Piazza: Q&A, Help, Announcements



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

## 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

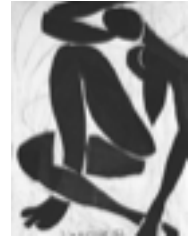
UC Berkeley "The Beauty and Joy of Computing" : Welcome, Abstraction (12)

# Abstraction & Detail Removal

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

## 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"



Naked Blue IV (Henri Matisse, 1952)

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

## Detail Removal



Automatic Generation of Detail Maps  
(Maneesh Agrawala @ UC Berkeley, among others)

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

## 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 often brought in to know what to remove and what to keep!



The London Underground 1928 Map & Harry Beck 1933 map.

UC Berkeley "The Beauty and Joy of Computing" : Welcome, Abstraction (16)

## Generalization

UC Berkeley "The Beauty and Joy of Computing" : Welcome, Abstraction (17)

## Generalization Example

- You have a farm with many different animals.
- Different food for each
- You have directions:
  - 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 (18)

## Generalization (in BJC) ... foreshadowing

- 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  
(Simply Scheme, Harvey)



## Abstraction Interfaces & Summary

## The Power of Abstraction, everywhere!

- Examples:

- Functions (e.g.,  $\sin x$ )
- Hiring contractors
- Application Programming Interfaces (APIs)
- Technology (e.g., cars)

*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

- Amazing things are built when these layer

- And the abstraction layers are getting deeper by the day!

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

## Summary

- Abstraction is one of the big ideas of computing!

- It's how mankind has engineered some of the greatest structures and managed the complexity



Someone who drove in 1930 could still drive a car today because they've kept the same

**Abstraction!**

(right pedal faster, left pedal slow)

- Two definitions

- Detail Removal
- Generalization

