



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/



BJC in one slide

- Big Ideas of Programming**
 - Abstraction
 - Algorithms (2)
 - Recursion (2)
 - Functions-as-data, λ , (2)
 - Programming Paradigms
 - Concurrency
 - Distributed Computing
- Big Ideas of Computing**
 - HowStuffWorks
 - 3D Graphics
 - Video Games
 - Computational Game Theory
 - Research Summaries
 - AI
 - HCI
 - Apps that Changed the World
 - Social Implications of Computing
 - Saving the World with Computing
 - How Twitter Works (guest lecture)
 - 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!!
 - One blog
 - Of students' own choice!!



Format & Textbooks

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

Mon	Wed	Fri
Lecture	Lecture	Discussion
Lab	Lab	
Lab	Lab	

- Selected Reading**

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

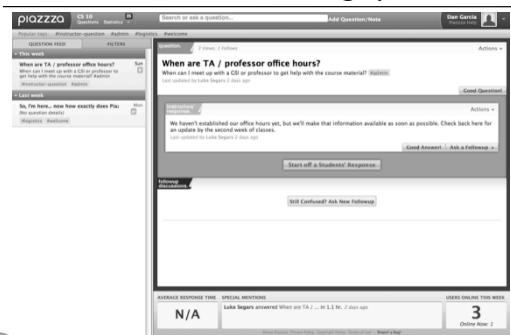


Peer Instruction

- 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



Piazza for {ask,answer}ing questions




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

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 (8)

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 (9)

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

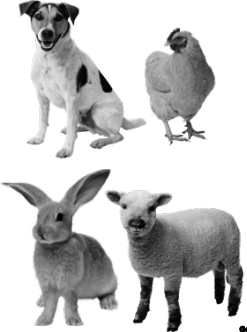


The London Underground 1928 Map & the 1933 map by Harry Beck.

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

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



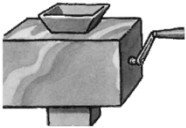
UC Berkeley "The Beauty and Joy of Computing": Welcome, Abstraction (11)

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

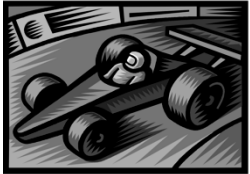


"Function machine" from *Simply Scheme* (Harvey)

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

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 died 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 (13)