Invite your friends to take CS10 next sem!

The Beauty and Joy of Computing

Michael Ball
Head TA

Calendar?

Slip days

Lecture #25
Summary & Review

Lab this week is Survey (0.20), online final (1.30)

Register iclicker, then turn in during lab or dis

BJC Art or Poem Submit this at final for extra credit!

OCULUS RIFT, NEXT "IT"?

Facebook's purchase of Oculus Rift is one indication that this is an incredibly HOT potential new technology. Gamers rejoice!

Discussion this week is important – course feedback + summary

oculusvr.com

Administrivia: Become active!

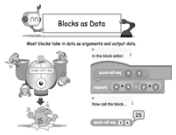
- With-Snap! Exam details
  - No exam handed out unless you've filled in both HKN + our survey
  - No "study sheets" needed / allowed since you have access to Snap!
- Final Exam details
  - Only bring pens, three 8.5"x11" handwritten sheets (writing on both sides).
  - Leave backpacks, books, calculators, cells & pagers home!
  - Everyone must take ALL of the final!
  - Bring your "Beauty and Joy of Computing" Art/Poem for extra credit!
- If you did well in CS10 and want to be on staff?
  - Usual path: **Lab Assistant ⇒ Reader ⇒ TA**
  - Indicate on your final survey whether you're even remotely interested
  - We strongly encourage anyone who gets an B or above in the class to follow this path…

Taking advantage of Cal Opportunities

- "The Godfather answers all of life's questions" – Heard in "You've got Mail"
- Why are we one of the top Universities in the WORLD?
  - Research, research, research!
  - Whether you want to go to grad school or industry, you need someone to vouch for you!
    - …as is the case with the Mob
- Techniques
  - Find out what you like, do lots of web research (read published papers), hit OH of Prof, show enthusiasm & initiative
- **`http://research.berkeley.edu/`**
- **`http://researchmatch.heroku.com/`**

Dan's Research Projects

- **CS Illustrated**



- **Improve CS10/Snap!**

- **Ensemble**

- **Improve Privacy Teaching**

We'll email class about opportunities this sprig…

Opportunities Next Semester

- CS61A (1st course in CS major)
  - Structure and Interpretation of Computer Programs, Python
- CS9 series (learn a second language)
  - I would recommend Python next, CS9H
- GamesCrafters DeCal (Game Theory R & D)
  - Develop SW, analysis on 2-person games of no chance. (e.g., go, chess, connect-4, nim, etc.)
  - Req: Game Theory / SW Interest
- MS-DOS X DeCal (Mac Student Developers)
  - Learn to program Macintoshes.
  - Req: Interest. Owning a mac helps, not required.
- UCBUGG DeCal (Recreational Graphics)
  - Develop computer-generated images, animations.
  - Req: 3D interest

Ok, I'm hooked! Where do I go next?

- **CS Major / Minor**
  - You are here
- **CS61A**
  - In Python, one big idea every week. Awesome!
- **CS61B**
  - In Java, data structures, algorithms and software engineering (lite)
- **CS61C**
  - In C and MIPS, Great ideas in computer architecture (parallelism) … I teach this!

Things to remember from CS10

- Abstraction
  - The key idea underpinning all computer science
  - …and (in CS10) functions, HOFs
- …From Blown to Bits
  - Technology has social implications (privacy, energy, copyright, etc); try to see the big picture
  - It also often has unintended consequences!
  - Things are never black or white, pure good or pure evil
- …From "Program or Be Programmed"
  - Technology has an explicit and implicit agenda, understanding it is important.
  - Learning to program is empowering (Steve Jobs' video)

The Future for Future Cal Alumni

- What's The Future?
- New Millennium
  - Always-on internet connectivity + internet of things!
  - AI breakthroughs
  - HCI breakthroughs
  - Post-PC Era (power is in cloud, interface in pocket)

"The best way to predict the future is to invent it" – Alan Kay

The Future is up to you!

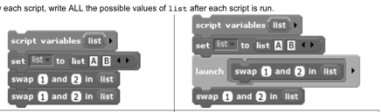
Question 11

Question 11: ***Eating my Halloween candy well beyond Thanksgiving…*** (6 pts)
Your mom asks you the difference between an iterative and recursive solution to a problem. You decide to explain it to her by showing how you would program a robot to eat a bag of M&Ms iteratively and recursively. Assume the robot knows how to "`eat one M&M`" and check if "`Bag is Empty`"

`Eat M&Ms Iteratively:`	`Eat M&Ms Recursively:`

Question 12: ***Magical Mystery Tour*** (11 pts)

a) Below each script, write ALL the possible values of `list` after each script is run.

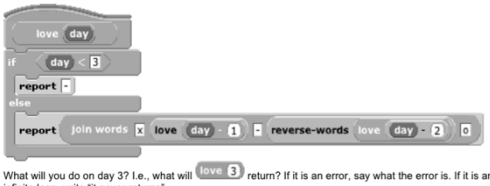


b) Assuming `swap` and `pick-random` are constant-time operations, `length-of(list)` is a linear-time operation, what is the running time of `mystery`? _____

c) What does `mystery` do? _____

d) Assuming you didn't know how `mystery` was written, but were just given the spec from your answer to (c), how would you test `mystery` really, really, really thoroughly? (this is called *black-box* testing)

Question 14: ***Give me some love! XOXO…*** (10 pts)
You decide to write `love`, a function to chart how affectionate you are (i.e., what you do) with your sweetie over the course of a given `day` (day 1 is your first day together, day 2 is your second, etc.). It returns a (possibly long) sentence whose elements are only: hugs (`o`), kisses (`x`), and just hanging out (`-`). We provide a helper block `reverse-words`, that does what you'd imagine: `reverse-words("CS10 is fun")` → `"fun is CS10"`



a) What will you do on day 3? I.e., what will `love 3` return? If it is an error, say what the error is. If it is an infinite loop, write "it never returns".

b) What will you do on day 4? I.e., what will `love 4` return? If it is an error, say what the error is. If it is an infinite loop, write "it never returns".

Question 14

c) Now let's do some analysis of your long-term relationship. What are the *first three and last three things you do on day 9999? That is,* what are the first three and last three letters of `love 9999`? Fill in the blanks.

_____ _____ _____ . . . _____ _____ _____

d) `love` can return a long and seemingly random sequence of `x`s, `o`s & `-`s. For each of the following activities, circle either POSSIBLE or IMPOSSIBLE if it's ever possible to do these things someday. The first one is already done for you.

- (POSSIBLE) IMPOSSIBLE : `"- - -"` (Hang out *three* times in a row)
- POSSIBLE IMPOSSIBLE : `"- - - -"` (Hang out *four* times in a row)
- POSSIBLE IMPOSSIBLE : `"o x"` (Hug immediately followed by a kiss)
- POSSIBLE IMPOSSIBLE : `"o o"` (Hug *twice* in a row)

Q15

Question 15: *A sorted Quest-ion* (10 pts)

Recall that our `map` function usually takes a `function` (of one argument) and one `list`, and applies the `function` to every element of the list, returning a `list` of the same size. It can also take *many* lists, and in this case the `function` must take the number of arguments equal to the number of lists, all the same size. The elements of the lists are one-by-one passed as arguments to the function. E.g.,

a) Write the following new block (a "cousin" of **`map`**) that takes a `function` (of two arguments) and a `list` (of at least two items) and applies the function to every *set of two neighbors*, returning a list one element smaller than the input. You may not use explicit recursion or iteration, and you may find the list manipulation helper functions (listed below) helpful. Here is an example call:


```
map(function)over-neighbors-in(list)
    report( _____ )
```

b) Now, using this function you've written in (a), write **`Sorted?(list)`** that returns when the input list is sorted in ascending order (i.e., every element is *smaller* than the ones after it, like the list `(3 6 7)`. Again, you may not use explicit recursion or iteration.

```
Sorted?(list)
    report( _____ )
```

List manipulation helper functions

Name	Description	Example
`item(num)of(list)`	Returns item at list index num	**`item(3)of( (cs10 is fun) )`** → `fun`
`all-but-first-of(list)`	Returns a new list with the last element removed. Doesn't change the original.	**`all-but-first-of( (cs10 is fun) )`** → `(is fun)`
`all-but-last-of(list)`	Returns a new list with the first element removed. Doesn't change the original.	**`all-but-last-of( (cs10 is fun) )`** → `(cs10 is)`

The Future for Future Cal Alumni

THANK YOU!

Good Luck!