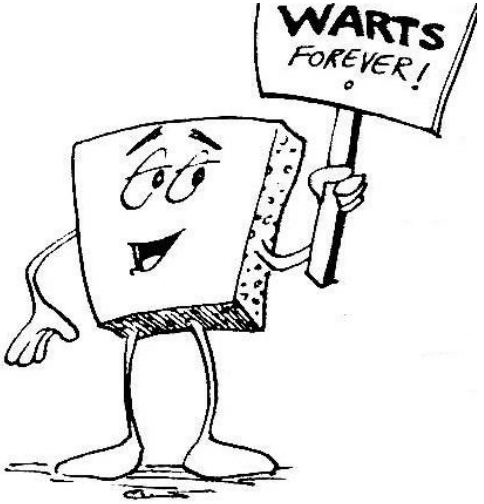
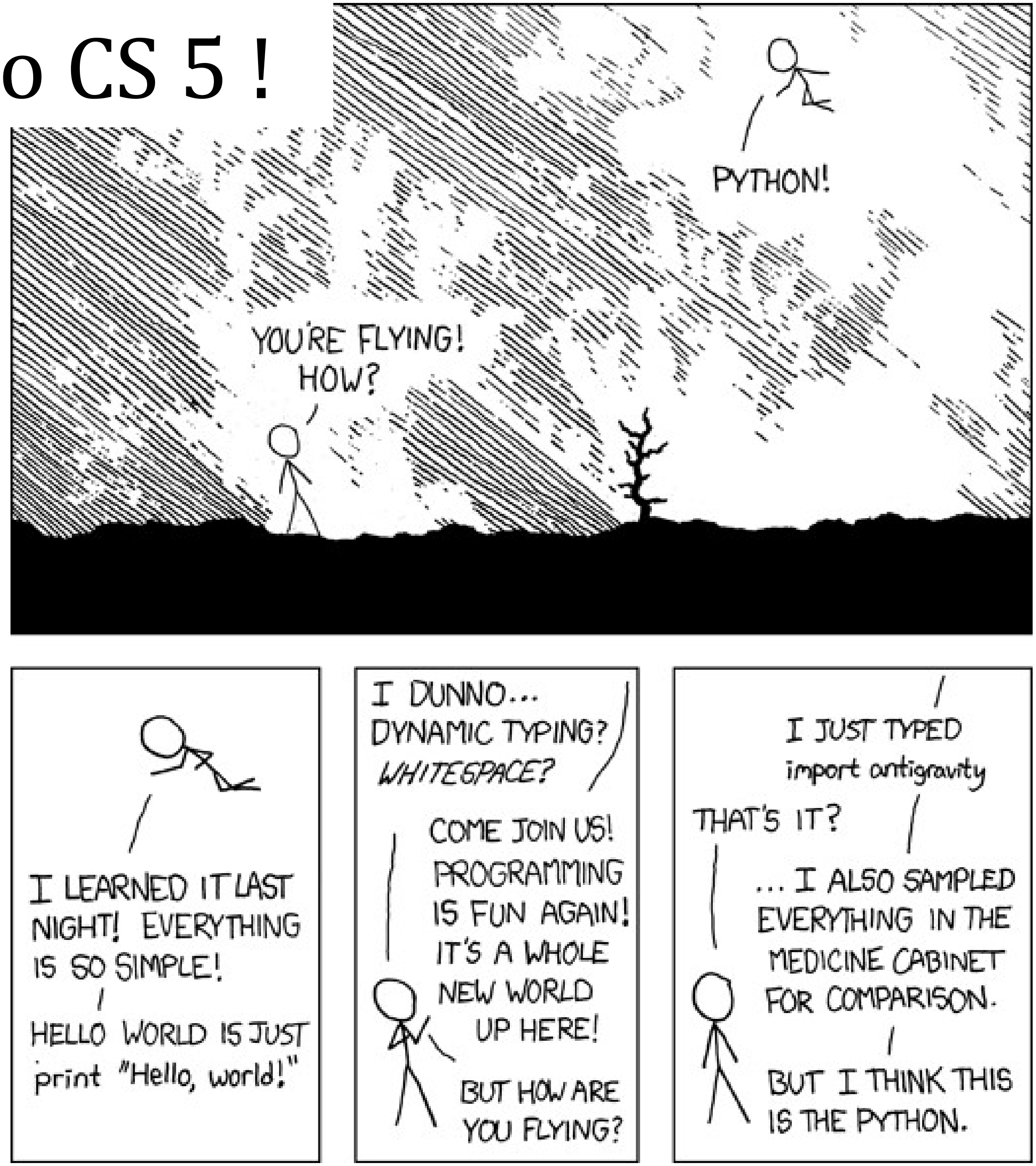
Welcome to CS 5 !



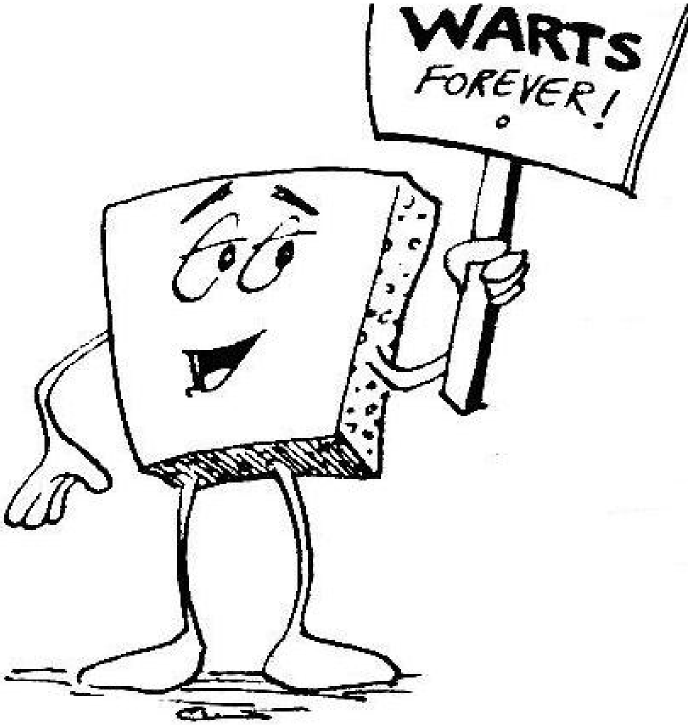

PYTHON!
YOU'RE FLYING! HOW?
I LEARNED IT LAST NIGHT! EVERYTHING IS SO SIMPLE!
HELLO WORLD IS JUST print "Hello, world!"
I DUNNO... DYNAMIC TYPING? WHITESPACE?
COME JOIN US! PROGRAMMING IS FUN AGAIN! IT'S A WHOLE NEW WORLD UP HERE!
BUT HOW ARE YOU FLYING?
I JUST TYPED import antigravity
THAT'S IT?
... I ALSO SAMPLED EVERYTHING IN THE MEDICINE CABINET FOR COMPARISON.
BUT I THINK THIS IS THE PYTHON.

xkcd, CS's id

Welcome to CS 5 !

Grab these lecture notes...

Introduction to CS



Wally Wart, a protrusive advocate of ***concrete*** computing

We don't have words strong enough to describe this class.

- US News and Course Report

Everyone will get out of this course – a lot!

- NYTimes Review of Courses

We give this course two thumbs…

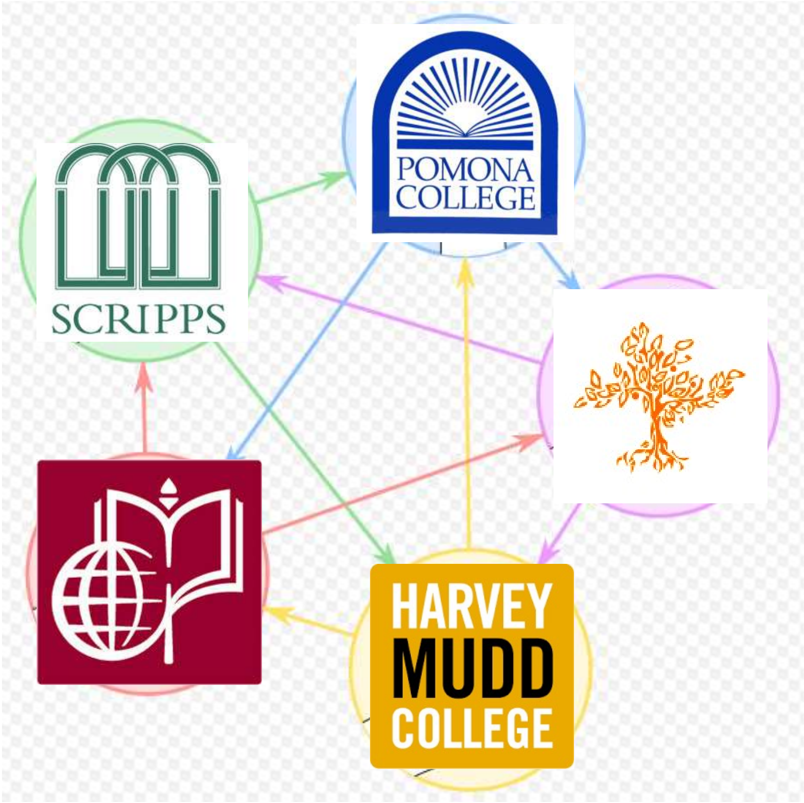
- *Meta*metacritic

1 handout…

slides & syllabus

official alien of CS 5 Gold

A word on 5 spots…



Welcome, not only to HMC, but *to all 5Cs*!

Introductions...

Zach Dodds
Olin B163 (HMC)
dodds@cs.hmc.edu

pursuer of *low-level* AI ➡

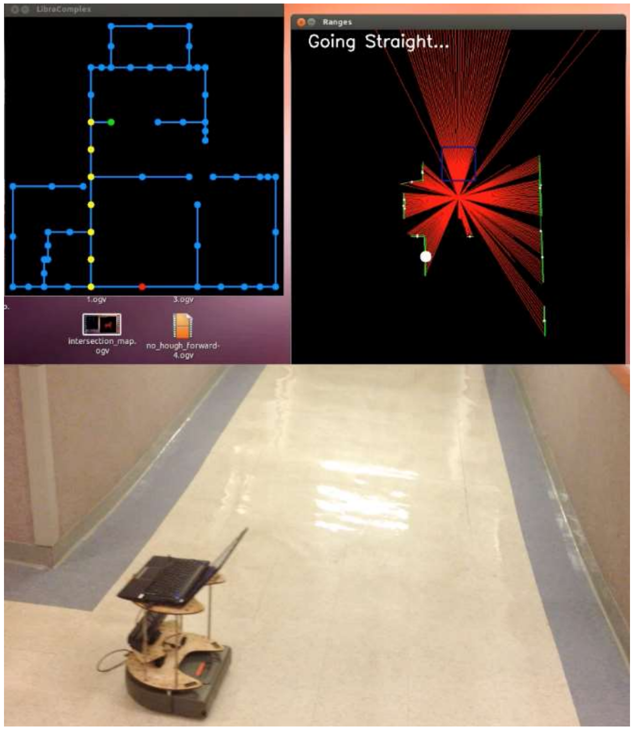
⬋ **taker of *low-quality* selfies**

fan of *low-tech* games ⬇

Speaking of
introductions

How I spend my summers ...?

actually, this "I" is
not quite accurate...



Robots

Chairs?

Outreach

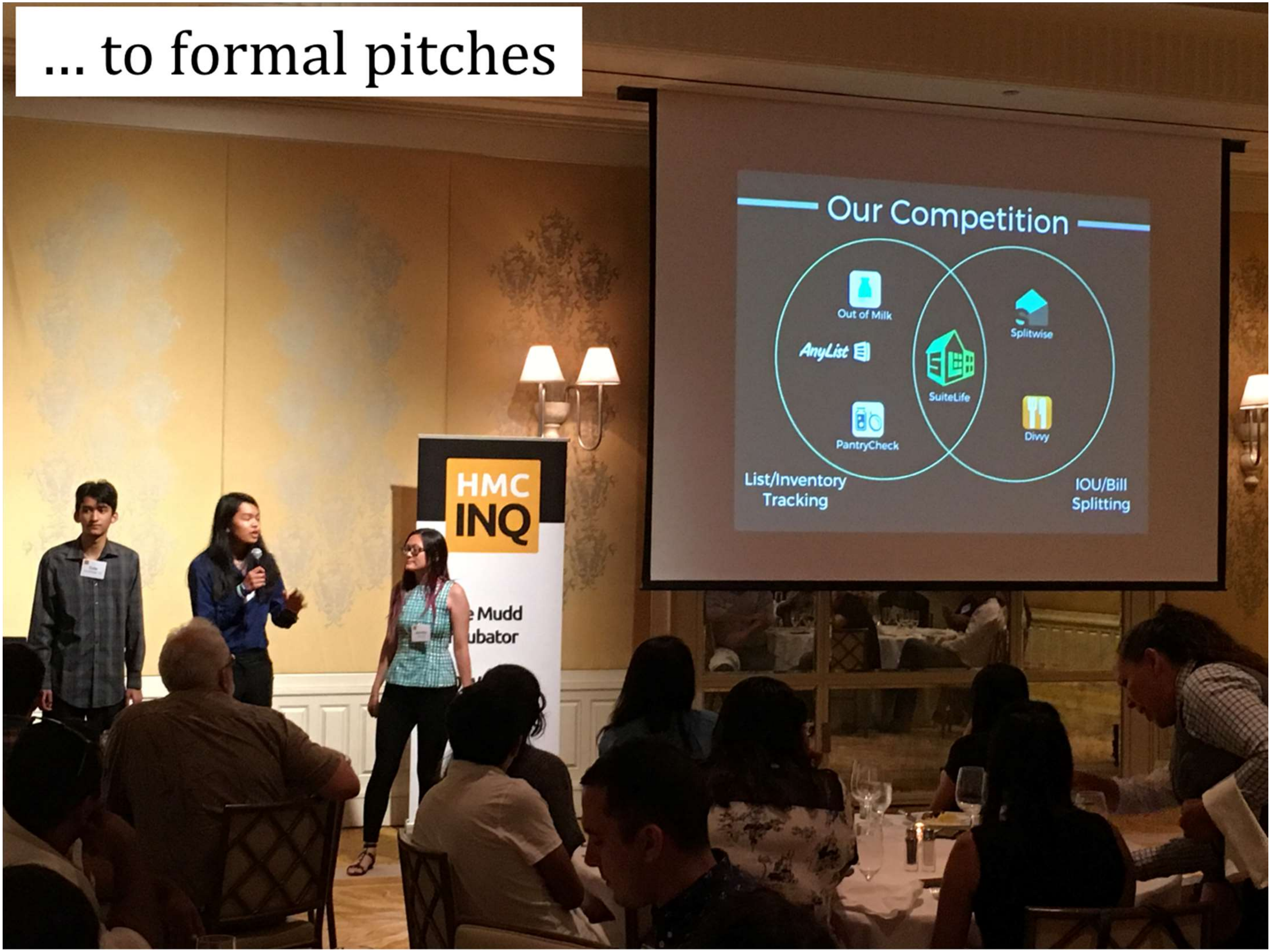
Who?!??

Dinos!

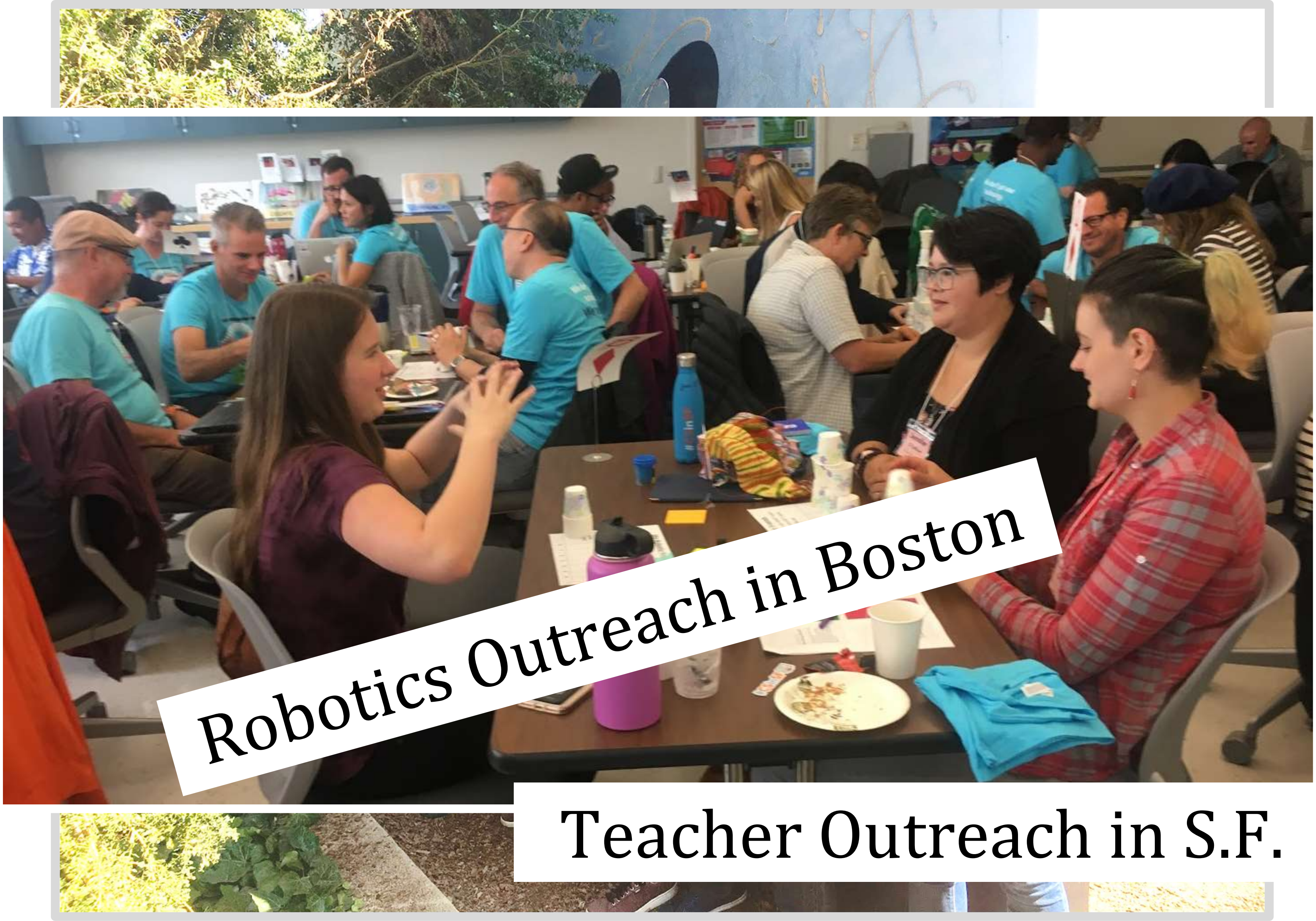
Algorithmic improvisation

Start-up ideas…

… to formal pitches



CS Staff: *Rising sophomores*, unite!



Robotics Outreach in Boston

Teacher Outreach in S.F.

CS Staff: *Rising sophomores*, unite!

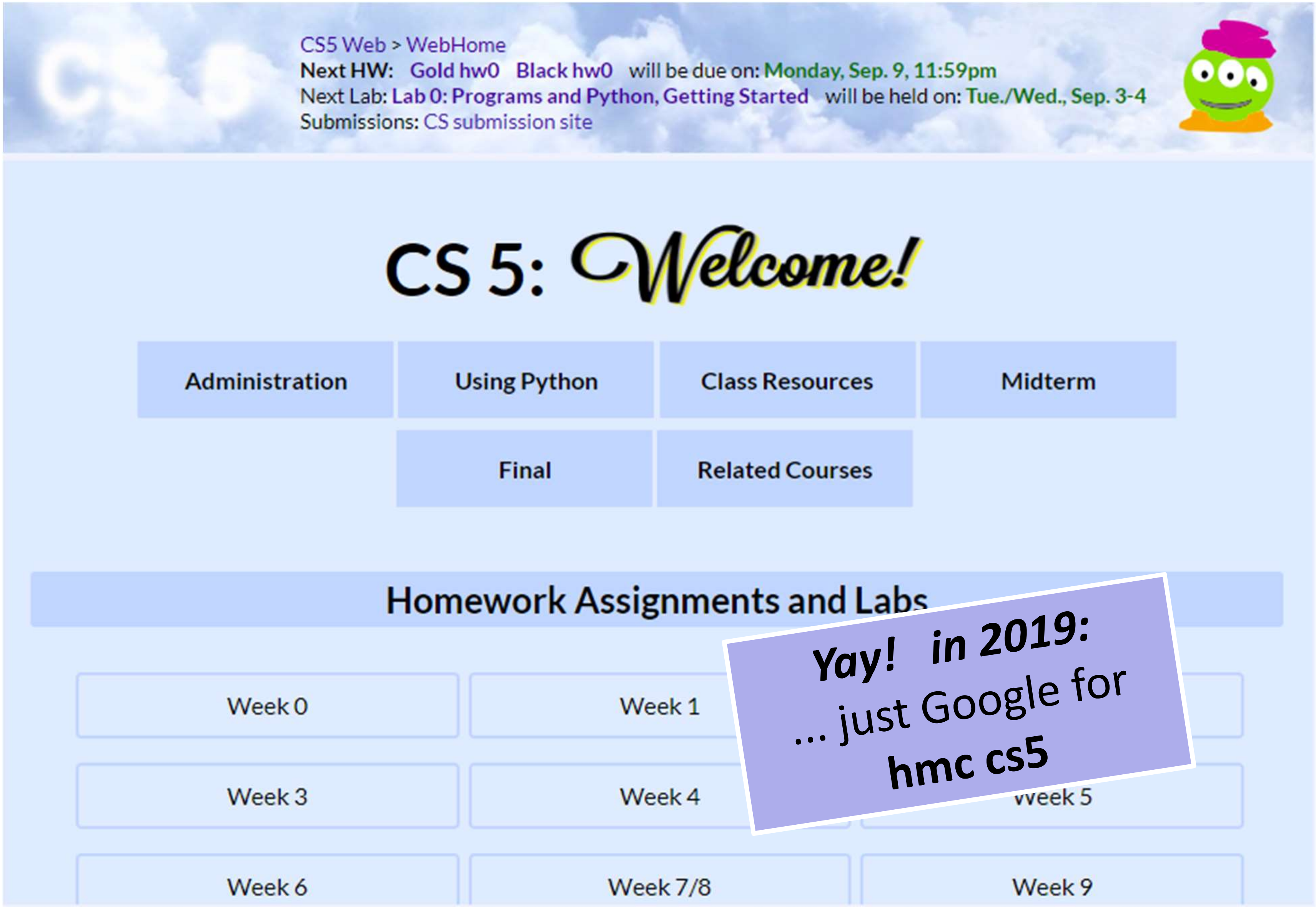
Where is this?

CS Staff: *Rising sophomores*, unite!

Lots of opportunities surrounding computing... (*at the 5Cs and beyond*)

Take-home message…

depending on where "home" is, perhaps…



CS5 Web > WebHome

Next HW: Gold hw0 Black hw0 will be due on: Monday, Sep. 9, 11:59pm

Next Lab: Lab 0: Programs and Python, Getting Started will be held on: Tue./Wed., Sep. 3-4

Submissions: CS submission site

CS 5: Welcome!

Homework Assignments and Labs

Yay! in 2019:

… just Google for

hmc cs5

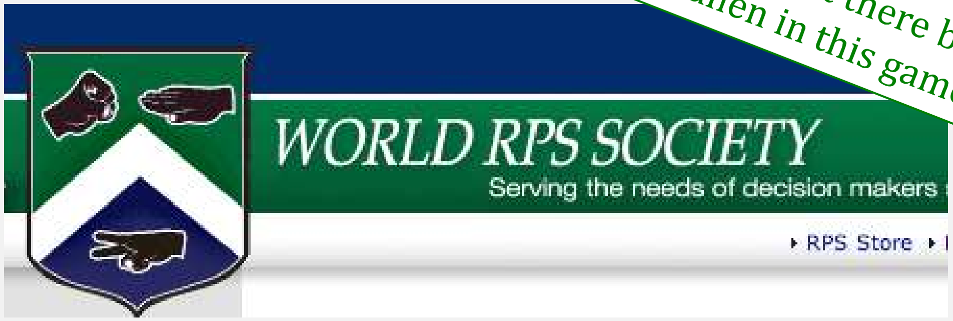
www.cs.hmc.edu/cs5

You're here ~ what's next?

1) How CS 5 runs…

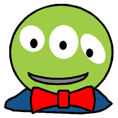
2) Python?!

the first Python HW
is ***choice!***



Shouldn't there be an alien in this game?

3) What ***is*** CS?



CS is just programming, right?

I'm not so sure…

Whatever it is,
it's definitely *alien*!

CS vs. programming ?

Spot the difference here?

```
print('hi')
```

```
print 'hi'
```

I *still* confuse these!

Spot the difference here?

```
print('hi')
```

python 3

```
print 'hi'
```

python 2

We'll be using python 3 this term...

Spot the difference here?

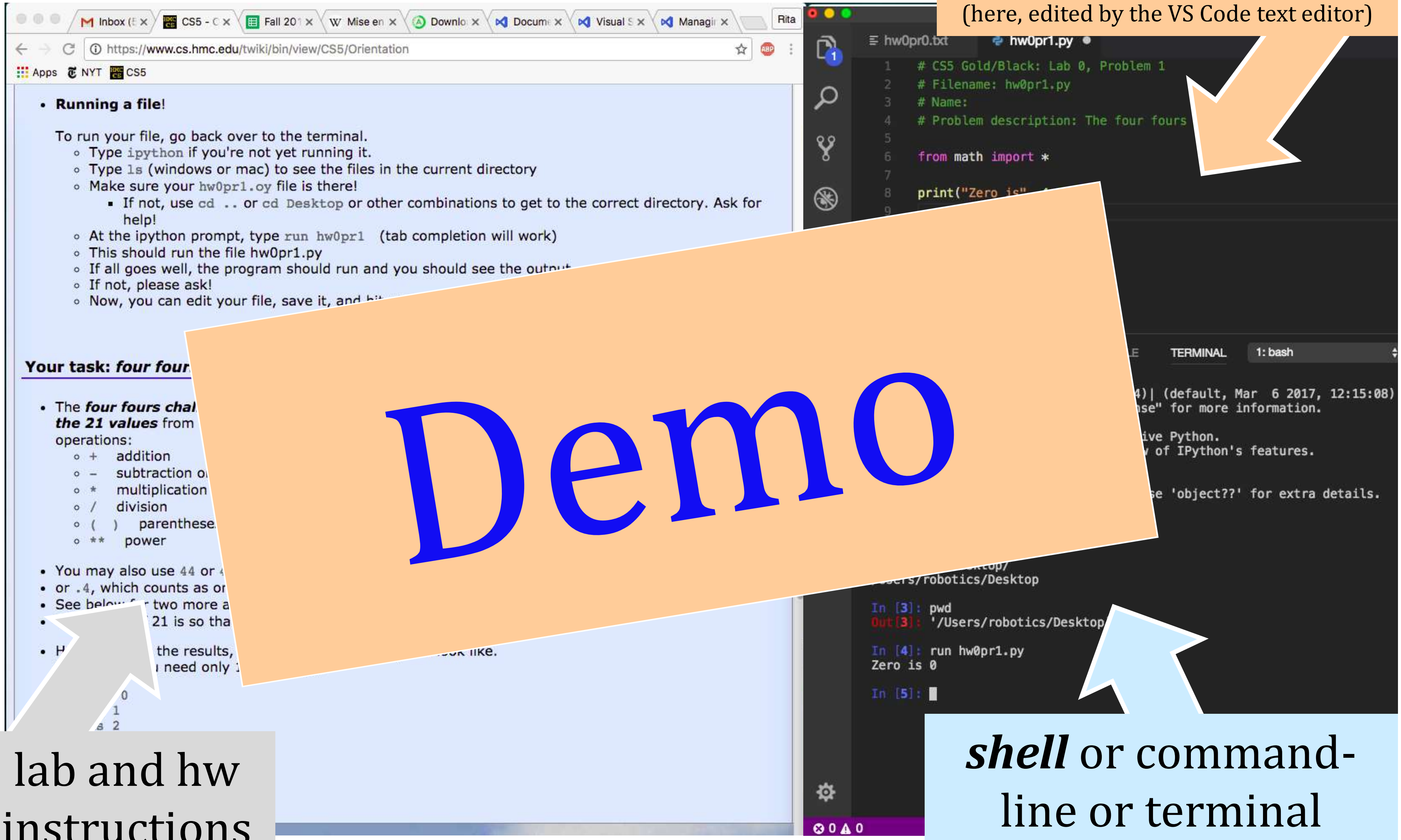
```
print('hi')
```

```
print 'hi'
```

Syntax!

We'll be using python 3 this term...

A *minute* of cs5 programming...



- **Running a file!**

  To run your file, go back over to the terminal.
  - Type ipython if you're not yet running it.
  - Type ls (windows or mac) to see the files in the current directory
  - Make sure your hw0pr1.oy file is there!
    - If not, use cd .. or cd Desktop or other combinations to get to the correct directory. Ask for help!
  - At the ipython prompt, type run hw0pr1   (tab completion will work)
  - This should run the file hw0pr1.py
  - If all goes well, the program should run and you should see the outpu
  - If not, please ask!
  - Now, you can edit your file, save it, and hi

Your task: *four four*

- The ***four fours chal*** ***the 21 values*** from operations:
  - + addition
  - - subtraction o
  - * multiplication
  - / division
  - ( ) parenthese
  - ** power
- You may also use 44 or 4
- or .4, which counts as or
- See below f r two more a
- 21 is so tha
- H the results, ook like. need only

```
0
1
s 2
```

```
# CS5 Gold/Black: Lab 0, Problem 1
# Filename: hw0pr1.py
# Name:
# Problem description: The four fours

from math import *

print("Zero is"
```

```
4)| (default, Mar  6 2017, 12:15:08)
se" for more information.

ive Python.
 of IPython's features.

se 'object??' for extra details.

ktop/
sers/robotics/Desktop

In [3]: pwd
Out[3]: '/Users/robotics/Desktop

In [4]: run hw0pr1.py
Zero is 0

In [5]: 
```

Python source code, a plain-text file

(here, edited by the VS Code text editor)

Demo

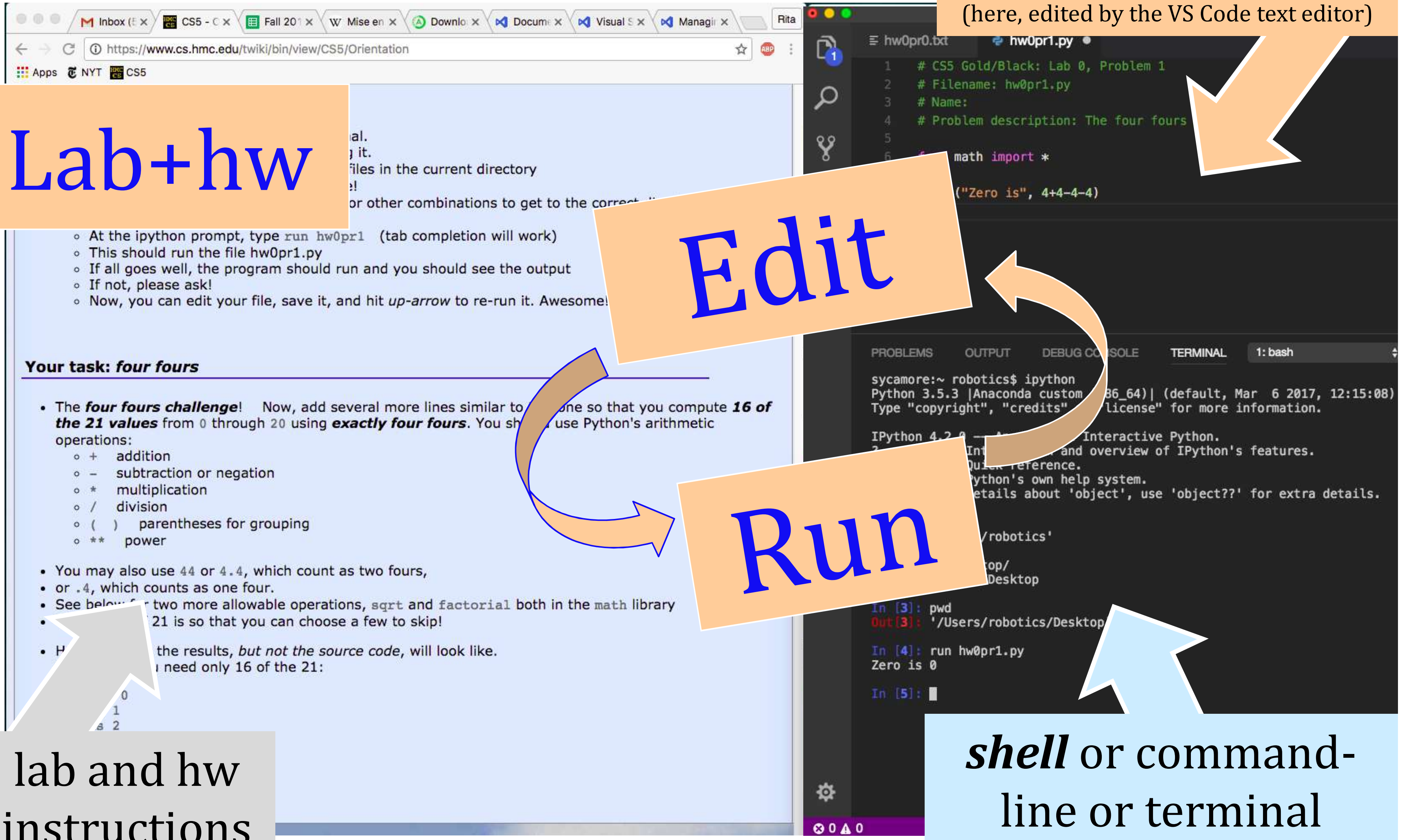
lab and hw instructions

shell or command-line or terminal

(the execution environment)

Lab 0: getting everything running ***on your own machine***

A *minute* of cs5 programming...



https://www.cs.hmc.edu/twiki/bin/view/CS5/Orientation

al.
g it.
files in the current directory
e!
or other combinations to get to the correct

- At the ipython prompt, type `run hw0pr1` (tab completion will work)
- This should run the file hw0pr1.py
- If all goes well, the program should run and you should see the output
- If not, please ask!
- Now, you can edit your file, save it, and hit *up-arrow* to re-run it. Awesome!

Your task: *four fours*

- The ***four fours challenge***! Now, add several more lines similar to one so that you compute ***16 of the 21 values*** from 0 through 20 using ***exactly four fours***. You sh use Python's arithmetic operations:
  - `+` addition
  - `-` subtraction or negation
  - `*` multiplication
  - `/` division
  - `( )` parentheses for grouping
  - `**` power
- You may also use `44` or `4.4`, which count as two fours,
- or `.4`, which counts as one four.
- See below two more allowable operations, `sqrt` and `factorial` both in the `math` library
- 21 is so that you can choose a few to skip!
- H the results, *but not the source code*, will look like. need only 16 of the 21:

```
0
1
s 2
```

hw0pr0.txt hw0pr1.py

```
# CS5 Gold/Black: Lab 0, Problem 1
# Filename: hw0pr1.py
# Name:
# Problem description: The four fours

f    math import *

     ("Zero is", 4+4-4-4)
```

```
sycamore:~ robotics$ ipython
Python 3.5.3 |Anaconda custom  86_64)| (default, Mar  6 2017, 12:15:08)
Type "copyright", "credits"    license" for more information.

IPython 4.2.0 -- An          Interactive Python.
?         Int          and overview of IPython's features.
          Quick reference.
          ython's own help system.
          etails about 'object', use 'object??' for extra details.

          /robotics'

          op/
          Desktop

In [3]: pwd
Out[3]: '/Users/robotics/Desktop'

In [4]: run hw0pr1.py
Zero is 0

In [5]: 
```

Python source code, a plain-text file

(here, edited by the VS Code text editor)

Lab+hw

Edit

Run

lab and hw instructions

shell or command-line or terminal

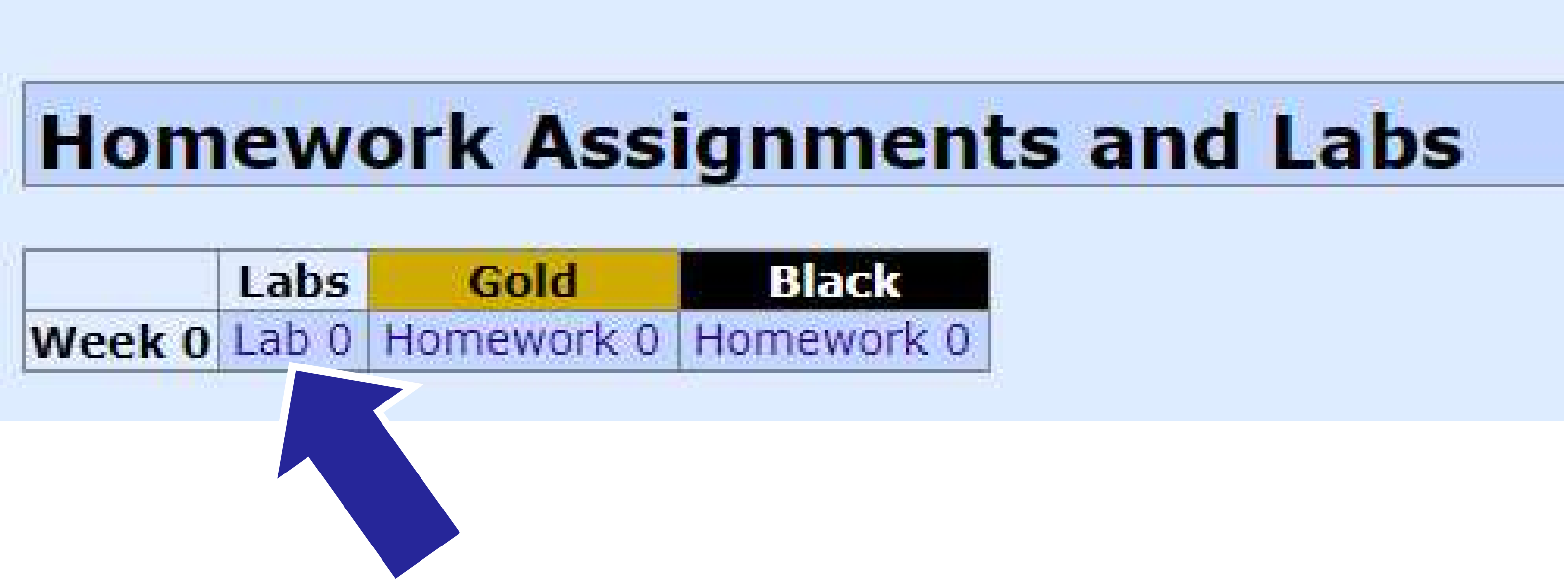
(the execution environment)

Lab 0: getting everything running ***on your own machine***

Lab 0: ***Happiness Suggestion***

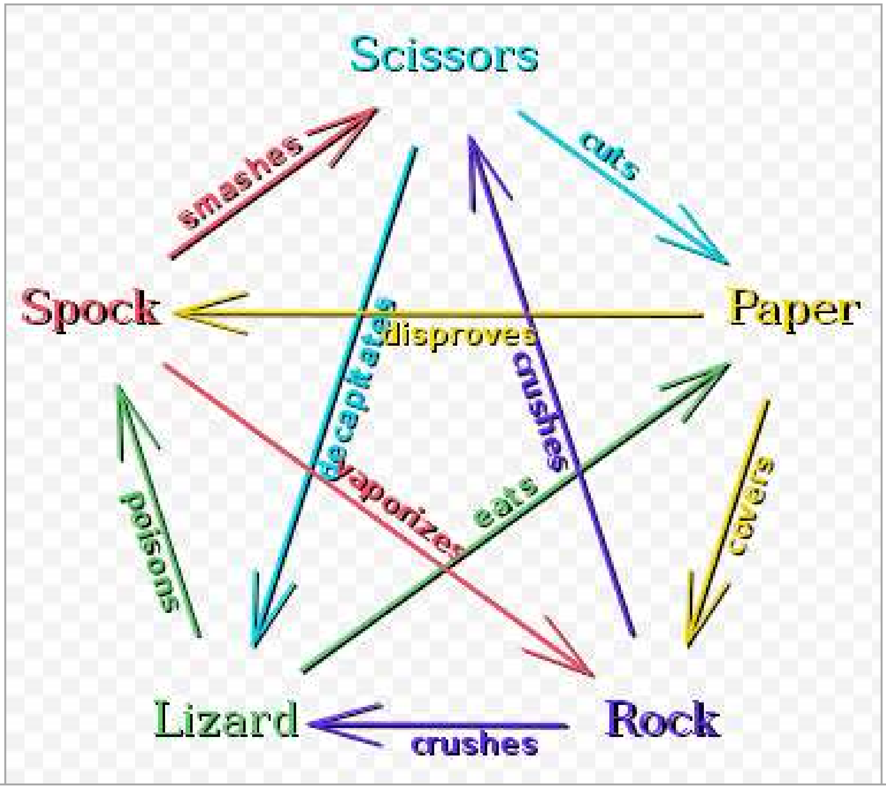
Download the software
BEFORE coming to lab:

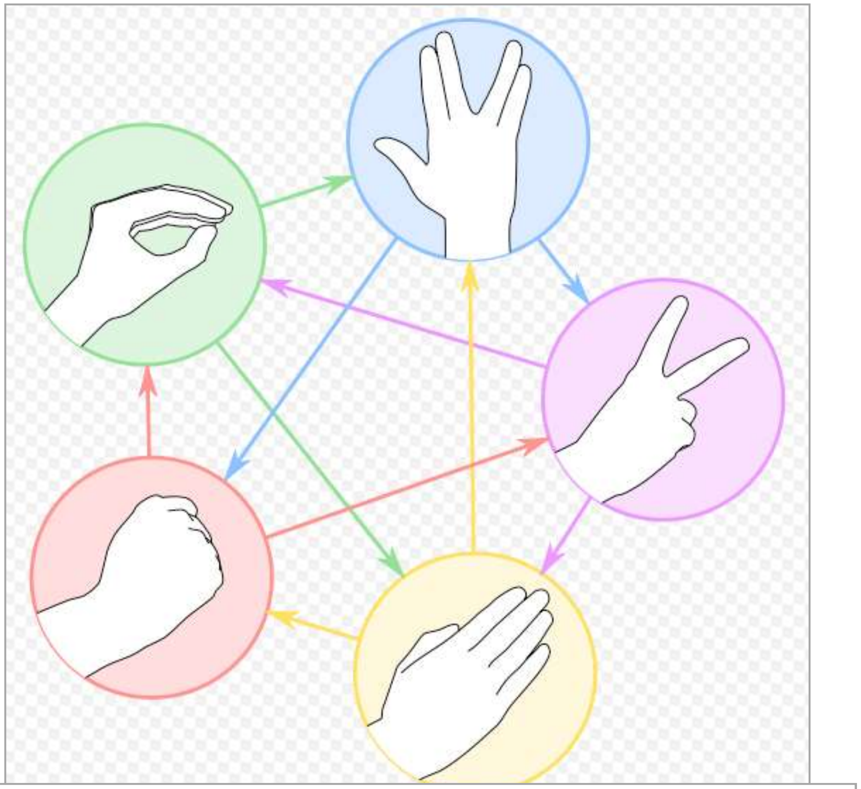
https://www.cs.hmc.edu/twiki/bin/view/CS5/OwnMachines



Homework Assignments and Labs

	Labs	Gold	Black
Week 0	Lab 0	Homework 0	Homework 0





rock – paper – scissors – lizard – Spock!

hw0: rock-paper-scissors

Let's play! Maybe two out of three?

Logically, I've got game!

http://www.youtube.com/watch?v=fqlDc2VICZ0 start at about :28

Soundbite Syllabus

Lectures

T and Th: *8:10-9:25 am*

Key

Come to Lectures!

Ins *y*, how)

We'd like to see you! Let me know if you'll be sick…

Lab

recommended by 4 out of 5 CS5 alums!

T or W: *2:45 - 4:45pm or 6-8 pm*

Gu

Come to Labs!

No *t for lab*

Will *SAVE* you time and effort in CS 5

Office hrs

F: 2:00-4:00 pm Linde Activities Center lab

Lots of help is available!

or, come to any of the *many* tutoring hrs.

HW

Monday

Hw is due on Monday nights…

Syllabus, briefly

Lectures

T and Th: *8:10-9:25 am*

Key skills, topics, and their motivation

Insight into the HW problems (what, ***why***, how)

We'd like to see you! Let me know if you'll be sick…

Lab

recommended by 4 out of 5 CS5 alums!

T or W: *2:45 - 4:45pm or 6-8 pm*

Guided progress on the week's hw

Not required, but encouraged: *full credit for lab*

Will ***SAVE*** you time and effort in CS 5

Office hrs

F: *2:30-4:30 pm, Linde Activities Center lab*

feel free to work on HW, to just stop by,
or, come to any of the ***many*** tutoring hrs!

HW

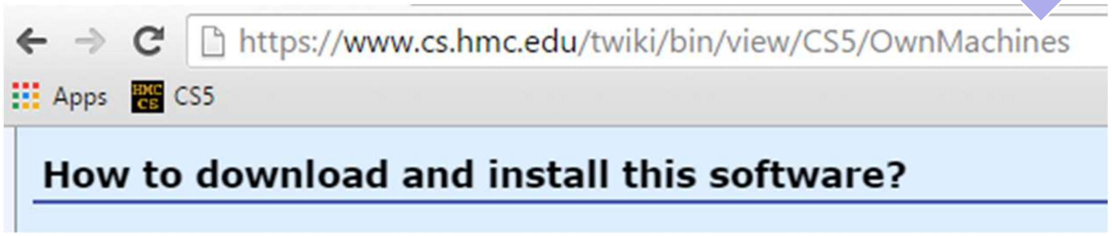
Monday nights: *due by 11:59 pm*

Each week's lab...

0) Find the lab! *Sign in...*

1) Get Python running...

download things ***now***, perhaps!



https://www.cs.hmc.edu/twiki/bin/view/CS5/OwnMachines

How to download and install this software?

demo

2) Edit, run, + submit a file...

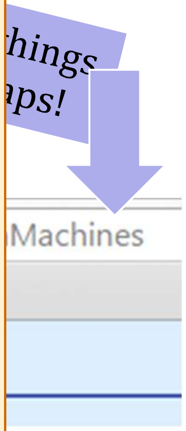
Encouraged: *bring your laptop*

Each week's lab...

Labs are <u>optional</u>, but *incentivized*.

If you come to lab, give a good-faith effort, and sign in, you'll receive **full credit for the lab problems** even if you don't finish

(you <u>do</u> need to submit by the usual hwk due date)



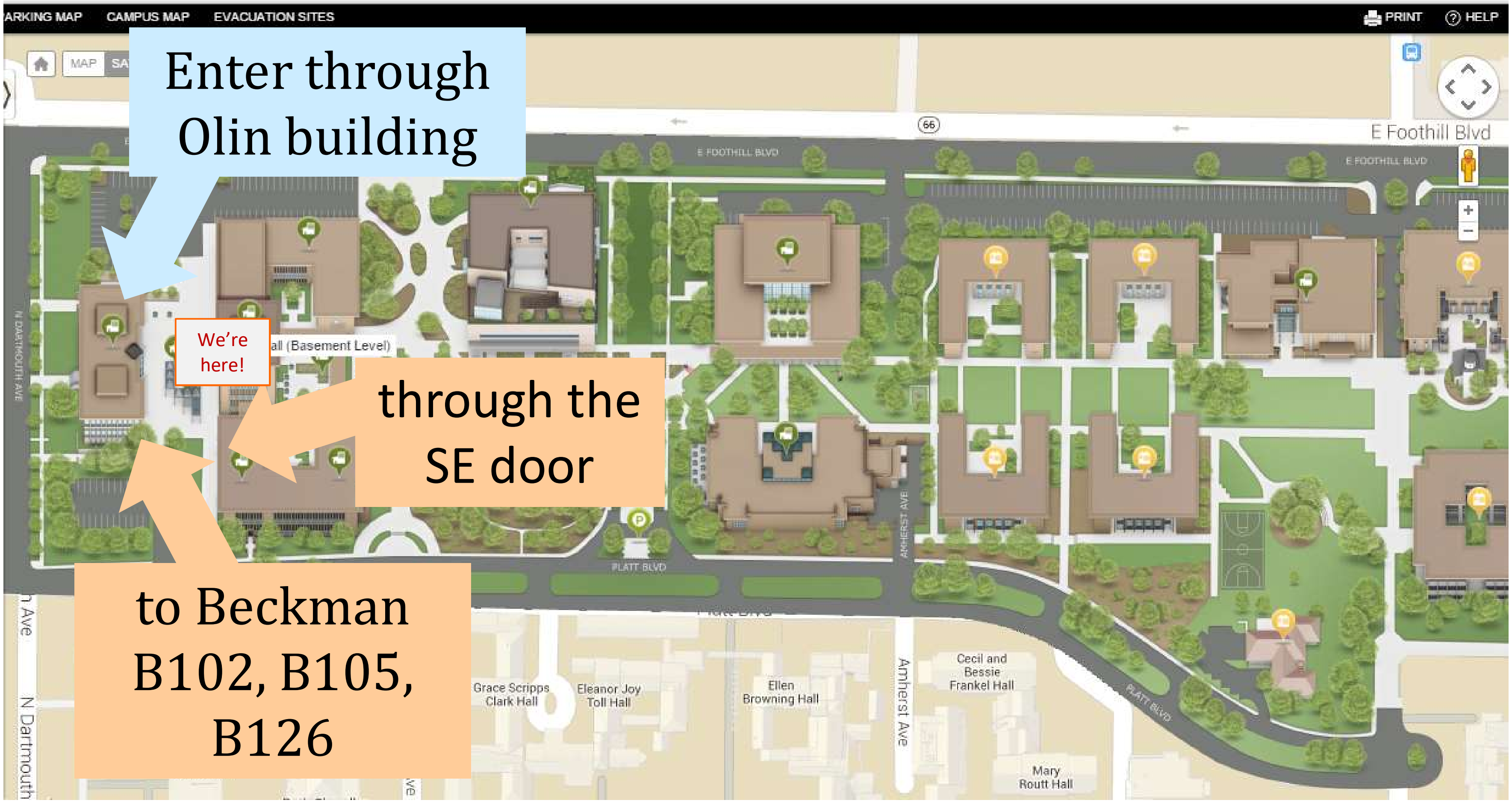
dem

.

Encouraged: *bring your laptop*

Evening lab?

Olin's Southeast door is open!



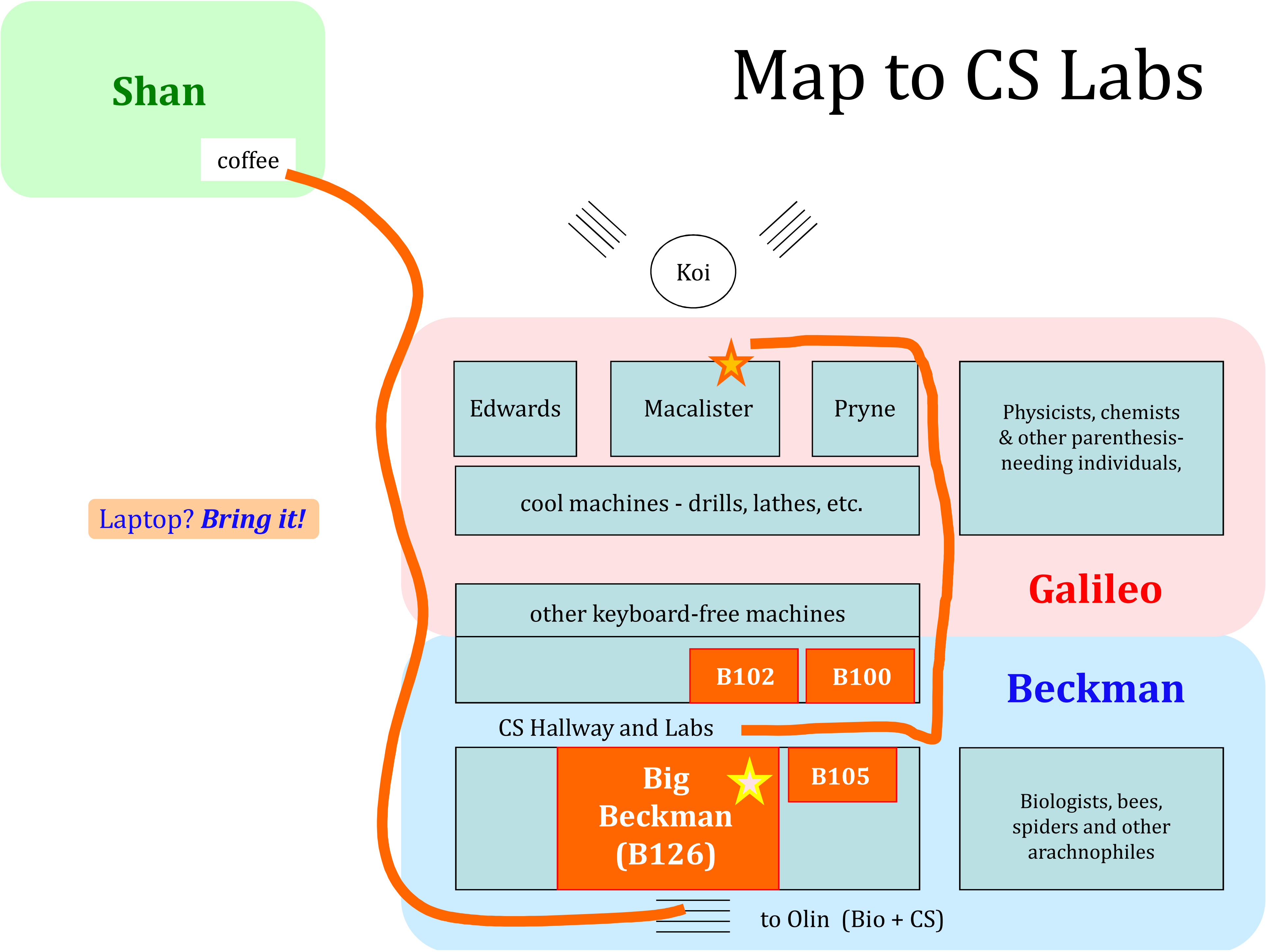
Enter through Olin building

We're here!

through the SE door

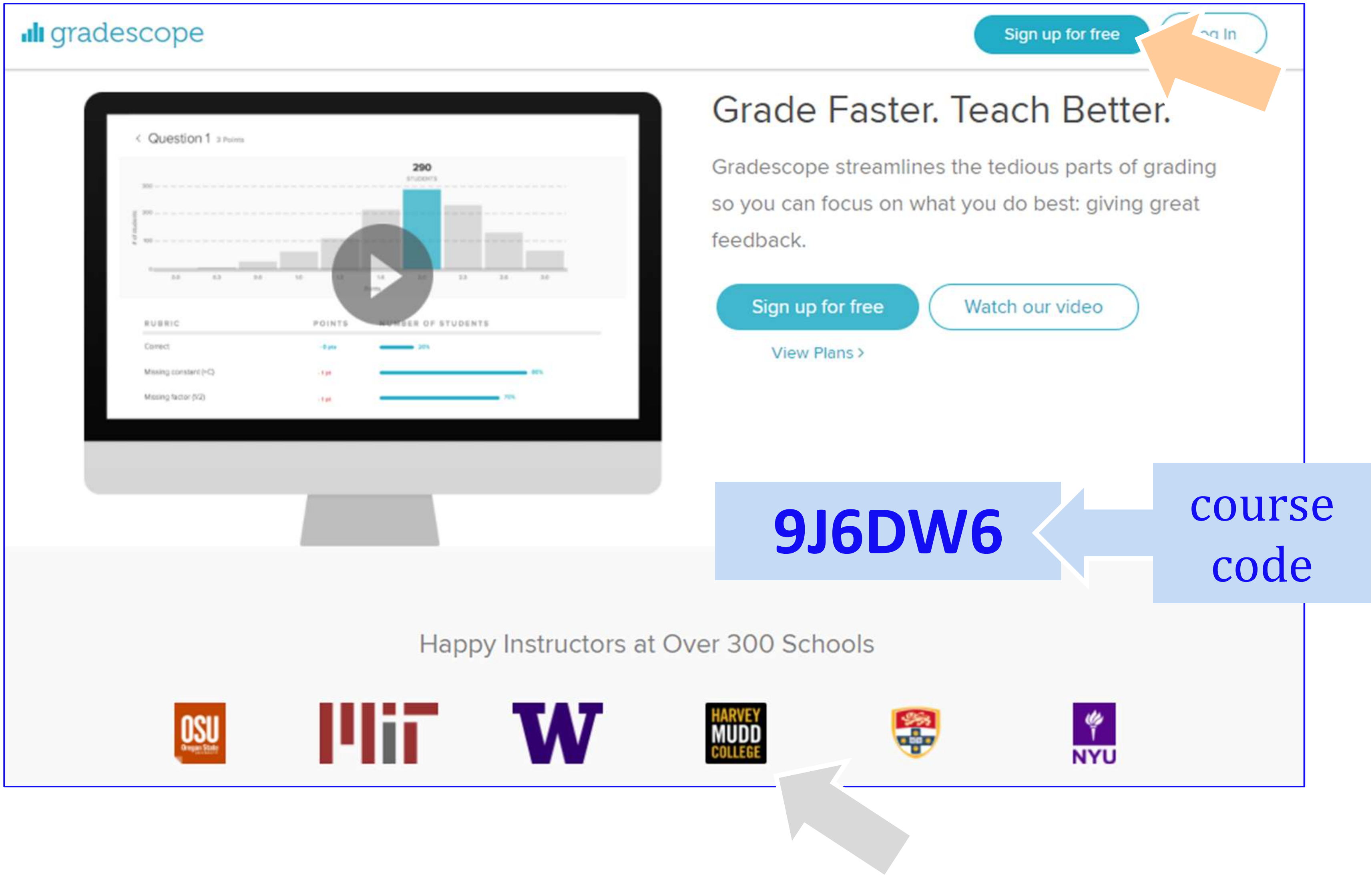
to Beckman B102, B105, B126

Map to CS Labs


Shan
coffee
Koi
Edwards
Macalister
Pryne
Physicists, chemists & other parenthesis-needing individuals,
cool machines - drills, lathes, etc.
Galileo
other keyboard-free machines
B102
B100
Beckman
CS Hallway and Labs
Big Beckman (B126)
B105
Biologists, bees, spiders and other arachnophiles
to Olin (Bio + CS)

Laptop? ***Bring it!***

Submissions: *GradeScope*



gradescope

Sign up for free

Log In

Grade Faster. Teach Better.

Gradescope streamlines the tedious parts of grading so you can focus on what you do best: giving great feedback.

Sign up for free

Watch our video

View Plans >

Happy Instructors at Over 300 Schools

9J6DW6

course code

Homework

Assignments

~ 5 problems/week

Due **Monday** evenings by 11:59 pm.

Extra credit is usually available…

You have 3 ***CS 5 Euros*** to use…

"Late Days"

Eur-ollowed to use one Euro for up to three hwks.

No need to let us know, even.

Collaborate!

Some problems are specified "individual-only."
Others offer the option of working as pairs/partners:

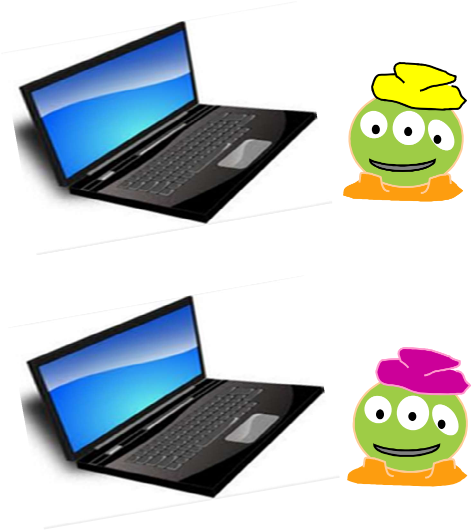
- You don't have to work in pairs/partners (that said, it's fun!)
- If you do, you must share the work equally - typing and coaching
- Be sure to indicate who your partner was at the submission site!

Pairs

one computer

tradeoff typing/debugging ~
about every 20 minutes

Partners



two computers

both partners type/debug ~
provide help as needed

Standard is the same either way: After finishing the hw, (a) *each person has contributed equally* and (b) *both could complete the problems on their own*

Submit with a partner as ***full co-owners*** of the work.

Honor Code

- You're *encouraged* to **discuss** problems with other students – or tutors - or any instructors.

- You may **not** share written, electronic or verbal solutions with other students, present or past:

Please ***do*** use the internet for Python language references.

Pleas ***do*** use other's eyes for finding syntax erorrs.

Do ***not*** use the internet (or intranet) to (try to) find solutions…

If you work as a pair/partners, the rules apply for the duo.

Even with three eyes, I need to borrow others' to find the syntax errors here!

Sign & submit CS's honesty policy ***online*** in this week's lab.

Grading

~ 65% Assignments

~ 30% Exams

~ 5% Participation/“quizzes”

```
if perc > .95:
    print('A')
elif perc > .90:
    print('A-')
elif perc > .70:
    print('Pass')
```

many take cs5 P/NC

see online syllabus for the full grade list...

Midterm? This feels more like a *2/3-term*!

Exams

Midterm	Th, Nov. 7, in-class
Final	Wed., Dec. 18

using a page of notes is OK on exams

the exams are *written*, not coded

the problems are modeled on the in-class "quizzes"

Choices, choices!

Let's **set** the value of **perc** to 0.91...

```
perc = 0.91

if perc > 0.95:
    print 'A'
elif perc > 0.90:
    print 'A-'
elif perc > 0.70:
    print 'Pass'
else:
    print 'Aargh!'
```

What will this program print, if **perc** is 0.91?

First – do you see the ***syntax errors*** here !?

What's here?

OCKS here:

ESTS here:

of **NTROL STRUCTURES** here:

Choices, choices!

Let's **set** the value of **perc** to 0.91...

```
perc = 0.91

if perc > 0.95:
    print('A')
elif perc > 0.90:
    print('A-')
elif perc > 0.70:
    print('Pass')
else:
    print('Aargh!')
```

Aargh!   ;-)

What will this program print,
if **perc** is 0.91?

Lots of Illuminating Solid
Parentheses!

What's here?

# of **BLOCKS** here:

# of **TESTS** here:

of **CONTROL STRUCTURES** here:

how many
tests are
executed?

Choices, choices!

Let's **set** the value of **perc** to 0.91...

```
perc = 0.91

if perc > 0.95:
    print('A')
elif perc > 0.90:
    print('A-')
elif perc > 0.70:
    print('Pass')
else:
    print('Aargh!')
```

What will this program print,
if **perc** is 0.91?

What's here?

# of **BLOCKS** here:

# of **TESTS** here:

of **CONTROL STRUCTURES** here:

how many
tests are
executed?

Choices, choices!

```
perc = 0.80

if perc > 0.95:
    print('A')
elif perc > 0.90:
    print('A-')
elif perc > 0.70:
    print('Pass')
else:
    print('Aargh!')
```

```
perc = 0.80

if perc > 0.00:
    print('Aargh!')
elif perc > 0.70:
    print('Pass')
elif perc > 0.90:
    print('A-')
else:
    print('A')
```

What does each of these programs print out, if `perc` is 0.8?

What value of `perc` gives an `'A-'` on the right?

How can you get a ***better*** grade on the right than the left?

Exclusive Choices

if … elif … else

```
if perc > 0.95:
    print('A')

elif perc > 0.90:
    print('A-')

elif perc > 0.70:
    print('Pass')

else:
    print('Aargh!')
```

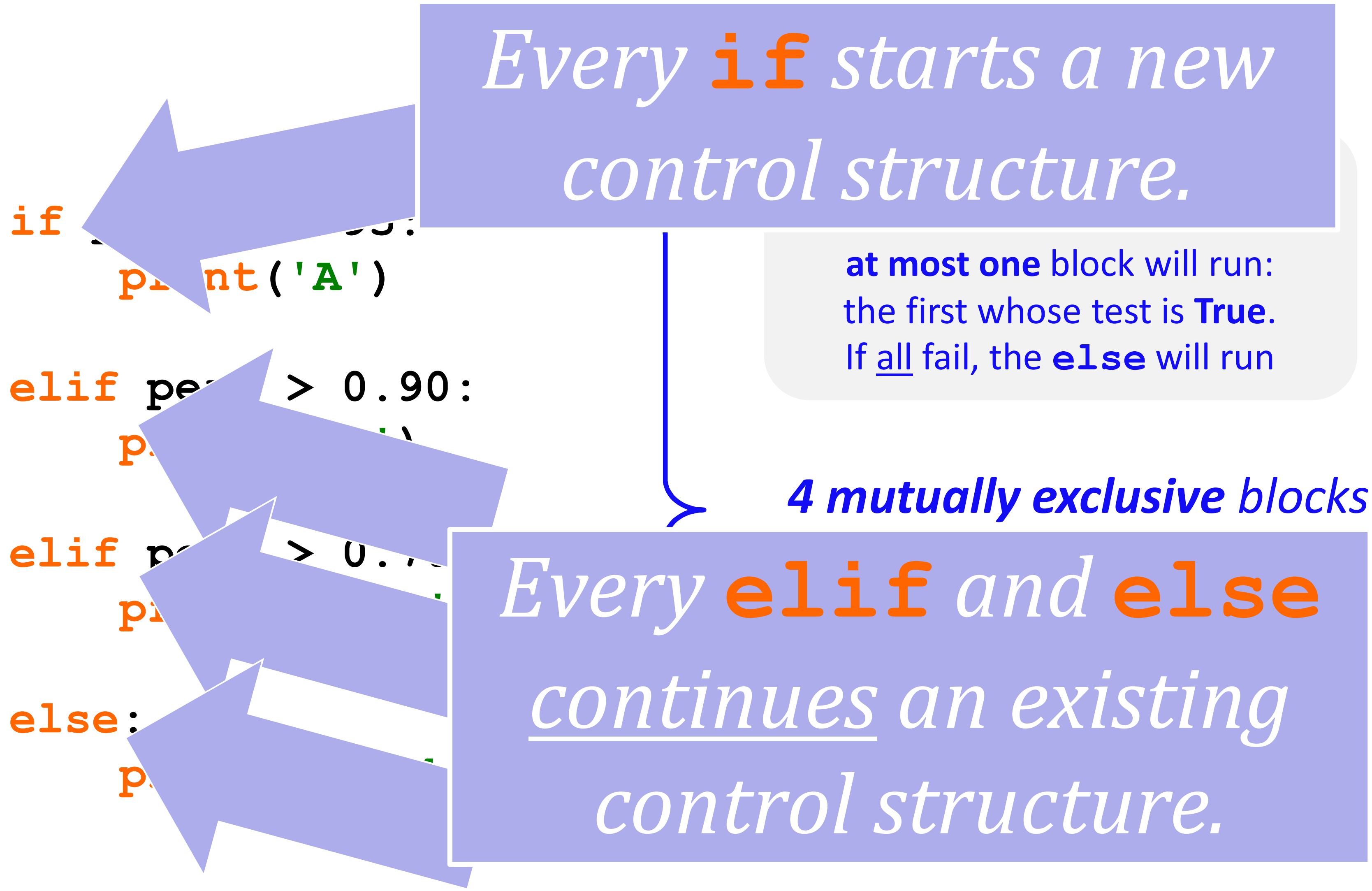
When using
`if . elif … . else`
at most one block will run:
the first whose test is **True**.
If all fail, the **`else`** will run

4 mutually exclusive *blocks*

in a single control structure

`elif` and `else` are optional

Exclusive Choices



Every **`if`** *starts a new control structure.*

```
if              5:
    pr   nt('A')

elif pe    > 0.90:
    p            ')

elif p     > 0.7
    p

else:
    p
```

at most one block will run:
the first whose test is **True**.
If all fail, the **`else`** will run

4 mutually exclusive *blocks*

Every **`elif`** *and* **`else`** *continues an existing control structure.*

`elif` and `else` are optional

What's the difference?

mutually exclusive blocks

nonexclusive blocks

What if **perc == .99** ? (How would we set it?)

How many separate ***control structures*** does each side have?

```
perc


if  perc > .95 :
     print('A')

elif  perc > .90 :
     print('A-')

elif  perc > .70 :
     print('Pass')
```

```
perc


if  perc > .95 :
     print('A')

if  perc > .90 :
     print('A-')

if  perc > .70 :
     print('Pass')
```

What's the difference?

mutually exclusive blocks

nonexclusive blocks

What if **perc == .99** ? (How would we set it?)

How many separate ***control structures*** does each side have?

```
perc = .99

if  perc > .95 :
     print('A')

elif  perc > .90 :
     print('A-')

elif  perc > .70 :
     print('Pass')
```

1

thing

```
perc = .99

if  perc > .95 :
     print('A')

if  perc > .90 :
     print('A-')

if  perc > .70 :
     print('Pass')
```

3

things

Nesting

for *decision-making*, we now have it ***all***…

Nesting

for *decision-making*, we now have it ***all***…

Nesting for *decision-making*, we now have it ***all***...

So, let's catch 'em ***all***...

Nesting

# Blocks ?

# Tests ?

# C. Structures ?

```
comp = 'rock'
user = 'paper'

if comp == 'paper' and user == 'paper':
    print('We tie. Try again?')

elif comp == 'rock':

    if user == 'scissors':
        print('I win! *_*')
    else:
        print('You win. Aargh!')
```

Does this program print the correct RPS result *this time*? *Does it* ***always****?*

Pair up with someone nearby – answer these questions together...

"Quiz"

Name ____________________

Your favorite _________ is ___________.

Your least favorite ___________ is ___________.

Name ____________________

Your favorite _________ is ___________.

Your least favorite ___________ is ___________.

What is something non-Claremont-collegey you have in common?

Then, try these Python q's:

(0) Find the 3 tests and 4 blocks here.

(1) What does this code print?

```
comp = 'rock'
user = 'rock'

if comp == 'rock':
   if user == 'paper':
      print('I win *_*!')
   elif user == 'scissors':
      print('You win.')
else:
   print('Tie.')
```

(2) As written, what output does this print?

```
comp = 'rock'
user = 'rock'

if comp == 'rock':
   print('I win *_*!')
if user == 'paper':
   print('You win.')
else:
   print('Tie: Ugh')
```

(3) ***Change*** these inputs to produce a completely correct RPS output here.

(4) How many of the 9 RPS **input cases** are *fully correctly* handled here?

	comp		
user	'rock'	'paper'	'scissors'
'rock'			
'paper'			
'scissors'			

(5) What is the ***smallest*** number of **blocks** and **tests** you'd need for a full game of RPS?

(**Extra**) What if it were RPS-5, which includes Lizard and Spock? How about RPS-101?

Pair up with someone nearby – answer these questions together...

"Quiz"

People

Name ________________

Your favo

Your least f ________.

Name ____________________

Your favorite __________ is ___________.

Your least favorite _______ .

Paper

What is son...ing non-Claremont-collegey you have in common?

Then, try these Python q's:

(0) Find the 3 tests and 4 blocks here.

(1) What does this code print?

Python

```
comp = 'rock'
user = 'rock'

if comp == 'roc
   if user == '
      print('I w      _ !')
   elif user == 'scissors':
      print('You win.')
else:
   print('Tie.')
```

(2) As written, what output does this print?

```
comp = 'rock'
user =    ck'

                'rock':
                win *_*!')
                paper':
                You win.')
else:
   print('Tie: Ugh')
```

(3) ***Change*** these inputs to produce a completely correct RPS output here.

(4) How many of the 9 RPS **input cases** are *fully correctly* handled here?

		comp		
		'rock'	'paper'	'scissors'
user	'rock'			
	'paper'			
	'scissors'			

(5) What is the ***smallest*** number of **blocks** and **tests** you'd need for a full game of RPS?

(**Extra**) What if it were RPS-5, which includes Lizard and Spock? How about RPS-101?

"Quiz"

Pair up with someone nearby – answer these questions together...

Name Zach Dodds

Your favorite tv show is Modern Family + Dr. Who

Your least favorite coffee is decaffeinated

Name T. E. Alien

ur favorite canned-meat food product is spam.

Your least favorite # is 41.999.

so close!

What is something non-Claremont-collegey you have in common? Our taste in hats!

en, try these Python q's:

(the 3 tests and 4 blocks here.

(1) What does this code print?

```
comp = 'rock'
user = 'rock'

if comp == 'rock':
   if user == 'paper':
      print('I win *_*!')
   elif user == 'scissors':
      print('You win.')
else:
   print('Tie.')
```

(2) As written, what output does this print?

```
comp = 'rock'
user = 'rock'

if comp == 'rock':
   print('I win *_*!')
if user == 'paper':
   print('You win.')
else:
   print('Tie: Ugh')
```

(3) ***Change*** these inputs to produce a completely correct RPS output here.

(4) How many of the 9 RPS **input cases** are *fully correctly* handled here?

user \ comp	'rock'	'paper'	'scissors'
'rock'			
'paper'			
'scissors'			

(5) What is the ***smallest*** number of **blocks** and **tests** you'd need for a full game of RPS?

(**Extra**) What if it were RPS-5, which includes Lizard and Spock? How about RPS-101?

Pair up with someone nearby – answer these questions together...

"Quiz"

Name ______________________

Your favorite __________ is ____________.

Your least favorite ____________ is ____________.

Name ______________________

Your favorite __________ is ____________.

Your least favorite ____________ is ____________.

What is something non-Claremont-collegey you have in common?

Then, try these Python q's:

(0) Find the 3 tests and 4 blocks here.

(1) What does this code print?

```
comp = 'rock'
user = 'rock'

if comp == 'rock':
   if user == 'paper':
      print('I win *_*!')
   elif user == 'scissors':
      print('You win.')
else:
   print('Tie.')
```

(2) As written, what output does this print?

```
comp = 'rock'
user = 'rock'

if comp == 'rock':
   print('I win *_*!')
if user == 'paper':
   print('You win.')
else:
   print('Tie: Ugh')
```

(3) ***Change*** these inputs to produce a completely correct RPS output here.

(4) How many of the 9 RPS **input cases** are *fully correctly* handled here?

		comp		
		'rock'	'paper'	'scissors'
user	'rock'			
	'paper'			
	'scissors'			

(5) What is the ***smallest*** number of **blocks** and **tests** you'd need for a full game of RPS?

(**Extra**) What if it were RPS-5, which includes Lizard and Spock? How about RPS-101?

Pair up with someone nearby – answer these questions togeth

Quiz"

Name ______________________

Your favorite ____ ________.

You ________.

Wha

Th

(0) Find t ... inputs ... letely ... here.

(1) What c ... 9 ... ully ... e?

```
comp = '
user = '

if comp =
   if use
      prin
   elif us
      print
else:
   print('T
```

...s the *smallest* number of **blocks** and **tests** you'd need for a full game of RPS?

(**Extra**) What if it were RPS-5, which includes Lizard and Spock? How about RPS-101?

'sors'

'paper'

'scissors'

In a moment... pass these up the aisles

(taking a picture, if you'd like)

... then, turn back to the notes

"Quiz" ~ problems 1+2

```
comp = 'rock'
user = 'rock'

if comp == 'rock':

    if user == 'paper':
        print('I win *_*!')
    elif user == 'scissors':
        print('You win.')

else:
    print('Tie.')
```

... what if this **else** block were indented?

"Quiz" ~ problems 3-5

```
comp = 'rock'
user = 'rock'

if comp == 'rock':
   print('I win *_*!')

if user == 'paper':
   print('You win.')

else:
   print('Tie: Ugh')
```

What does this program print?

"Quiz" ~ problems 3-5

```
comp = ~~'rock'~~
user = ~~'rock'~~

if comp == 'rock':
    print('I win *_*!')

if user == 'paper':
    print('You win.')

else:
    print('Tie: Ugh')
```

		comp		
		'rock'	'paper'	'scissors'
user	'rock'			
	'paper'	?	?	?
	'scissors'			

How many possible "input cases" are there?

For how many is this program correct?

*How **efficient** can we be?*

For RPS-3? RPS-5? RPS-101?

"Quiz" ~ problems 3-5

```
comp = ~~'rock'~~
user = ~~'rock'~~

if comp == 'rock':
   print('I win *_*!')

if user == 'paper':
   print('You win.')

else:
   print('Tie: Ugh')
```

	comp		
user	'rock'	'paper'	'scissors'
'rock'	I win *_*! Tie: Ugh	Tie: Ugh	Tie: Ugh
'paper'	I win *_*! You win.	You win.	You win.
'scissors'	I win *_*! Tie: Ugh	Tie: Ugh	Tie: Ugh *correct!*

How many possible "input cases" are there?

For how many is <u>this</u> program correct?

*How **efficient** can we be?*

For RPS-3? RPS-5? RPS-101?

"Quiz" ~ problems 3-5

```
comp = ~~'rock'~~
user = ~~'rock'~~

if comp == 'rock':
   print('I win *_*!')

if us
   pr

else
   print('Tie: Ugh'
```

	comp 'rock'	'paper'	'scissors'
user 'rock'	I win *_*! Tie: Ugh	Tie: Ugh	Tie: Ugh
	_! win.	You win.	You win.
	_! Ugh	Tie: Ugh	Tie: Ugh *correct!*

A correct RPS is possible with <u>only</u> **if** ... **elif** ... **else** *!*

How many possible "input cases" are there?

For how many is <u>this</u> program correct?

*How **efficient** can we be?*

For RPS-3? RPS-5? RPS-101?

Quiz"

Pair up with someone nearby – answer these questions togeth

Name ______________________

Your favorite ______ ____.

You ____.

Wha

Th

(0) Find t ... inputs ... letely ... here.

(1) What c ... 9 ... fully ... e?

```
comp = '
user = '

if comp =
    if use
        prin
    elif us
        print
else:
    print('T
```

Ok! Pass these to the aisles + "upward"

(take a picture, if you'd like)

... then, turn back to the notes

... is the ***smallest*** number of **blocks** and **tests** you'd need for a full game of RPS?

(**Extra**) What if it were RPS-5, which includes Lizard and Spock? How about RPS-101?

	sors'		
'			
'paper'			
'scissors'			

CS != programming

programming : **CS** ::

longboards : HMC maybe 5Cs?

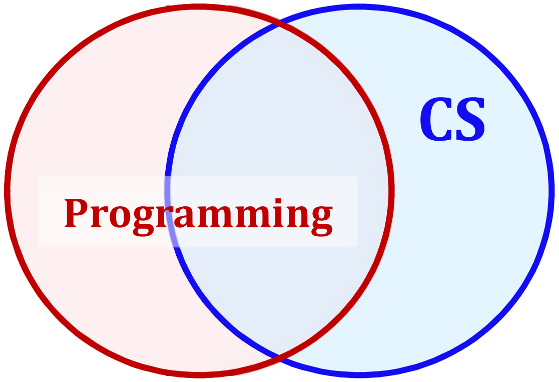
capital : business venture

equations : mathematics

language : ideas

web search : knowledge

Tesla : Google



programs are a ***vehicle***, but not the destination

CS != programming

"not equal to"

CS != programming

So, what is CS?

Punctuation matters!
So what? *is* CS

Today in CS5

1) How CS 5 runs…

2) Python?!

Shouldn't there be an alien in this game?

3) What *is* CS?

CS is just programming, right?

I'm not so sure…

Whatever it is, it's definitely *alien*!

What is CS a science *of* ?

the study of ***complexity***:

*How can **it** be done?*

*How well can **it** be done?*

*Can **it** be done at all?*

***it* ~ information**

or, more precisely, *a process transforming information from one form to another*

3 examples?
That's ***it*** for me!

We'll look at 3 examples – each of which you'll ***construct*** in CS 5 ...at least to some extent!

What is CS?

*How can **it** be done?*

*How well can **it** be done?*

*Can **it** be done at all?*

Can you solve the problem?

Can you create a *process* to solve such problems?

'HUMAN'

'CHIMPANZEE'

What is the ***Longest Common Subsequence*** between 2 strings?

biology's string-matching problem, "LCS"

'CGCTGAGCTAGGCC...'

'ATCCTAGGTAACTG...'

+10^9more

Eye oneder if this haz othur aplications?

What is CS?

Feels like home!

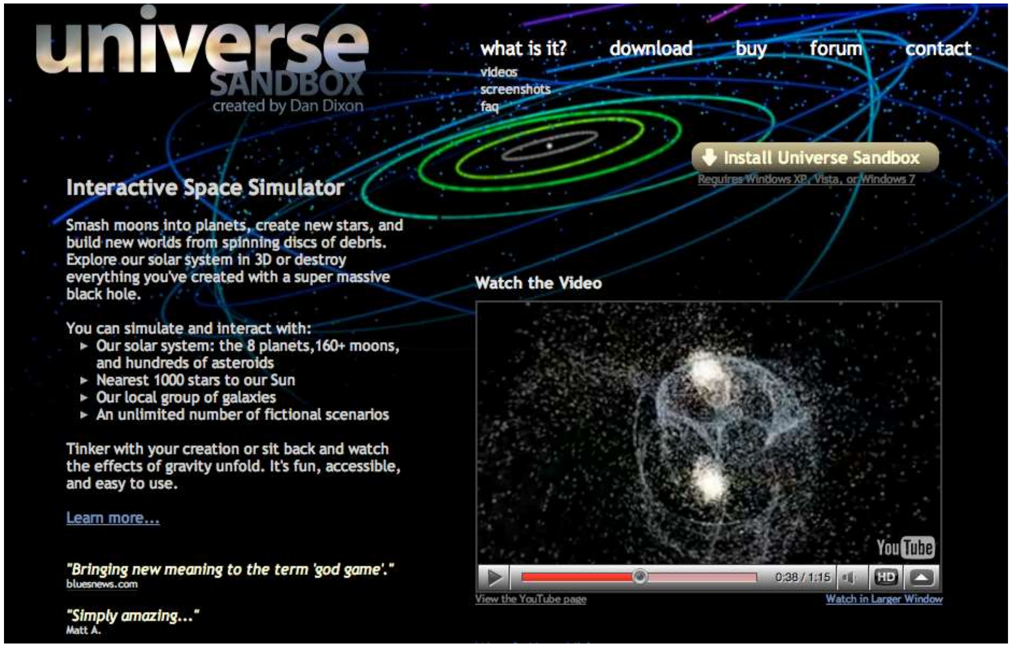
*How can **it** be done?*

*How well can **it** be done?* →

*Can **it** be done at all?*

How *quickly* can you find a solution?

Is your solution the *"best"* possible?



Interactive Space Simulator

Smash moons into planets, create new stars, and build new worlds from spinning discs of debris. Explore our solar system in 3D or destroy everything you've created with a super massive black hole.

You can simulate and interact with:

- Our solar system: the 8 planets,160+ moons, and hundreds of asteroids
- Nearest 1000 stars to our Sun
- Our local group of galaxies
- An unlimited number of fictional scenarios

Tinker with your creation or sit back and watch the effects of gravity unfold. It's fun, accessible, and easy to use.

Learn more...

"Bringing new meaning to the term 'god game'."
bluesnews.com

"Simply amazing..."
Matt A.

Watch the Video

How much work is needed to simulate N stars?

chemistry's + physics's "N-body" problem

What if N is a million-and-one...?

What is CS?

*How can **it** be done?*

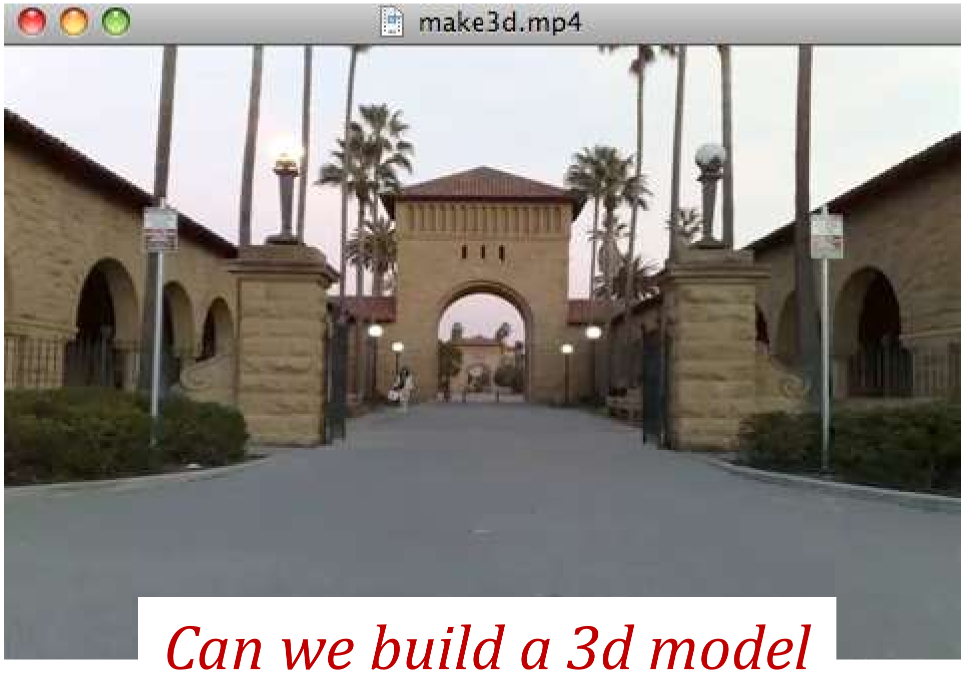
*How well can **it** be done?*

*Can **it** be done at all?*

Is your problem *solvable*?

How can you tell !?

many problems are *uncomputable*…
… and you'll *prove* this!



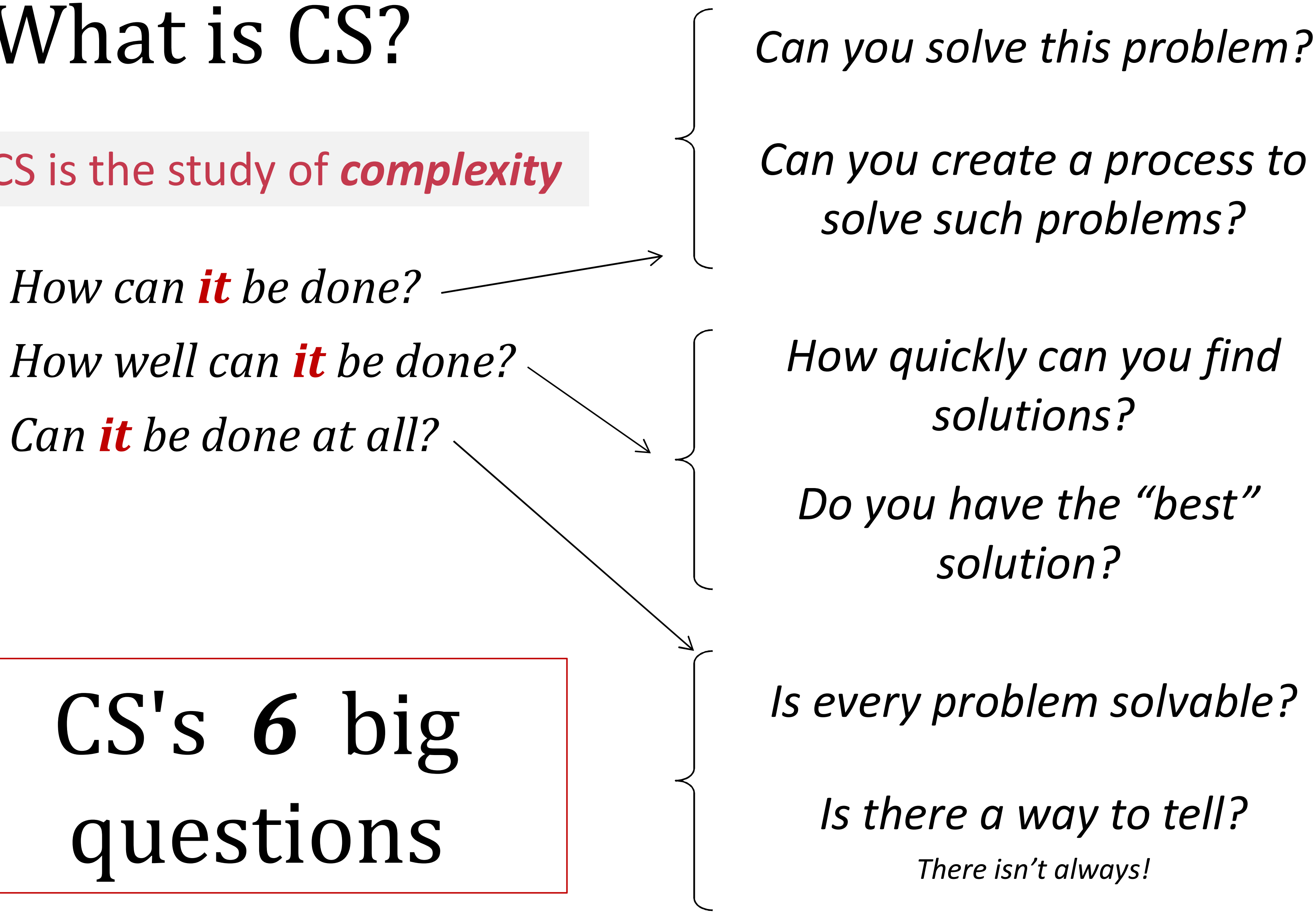
Can we build a 3d model from one 2d image?

Andrew Ng's "Make3d"

All three eyes tell me that Make3d
has just failed ~ epically!

What is CS?

CS is the study of ***complexity***



*How can **it** be done?*

*How well can **it** be done?*

*Can **it** be done at all?*

Can you solve this problem?

Can you create a process to solve such problems?

How quickly can you find solutions?

Do you have the "best" solution?

Is every problem solvable?

Is there a way to tell?

There isn't always!

CS's ***6*** big questions

Only one is programming. *Which one?*

What is CS?

CS is the study of ***complexity***

*How can **it** be done?*

*How well can **it** be done?*

*Can **it** be done at all?*

Can you solve this problem? CS

Can you create a process to solve such problems? ***programming*** + CS

How quickly can you find solutions? CS

Do you have the "best" solution? CS

Is every problem solvable? CS

Is there a way to tell? CS

There isn't always!

CS's ***6*** big questions

Only <u>one</u> is programming. *Which one?*

CS’s – and CS5's – philosophy:

Whatever you are,
be a good one.

- Abraham Lincoln

More and more,
CS can help!

*Remember ~ **Lab** this week*

Tue. or Wed. ~ afternoon or evening

Bring your laptop to Beckman B126 (here)

- *or* use one of the CS machines in B105/B102

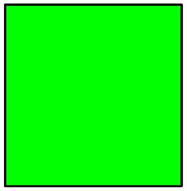
Get started with Python/text editor/cmdline…

See you in lab!

(*perhaps at 2:44:44 today…?*)

though it's more than a few bits early!

Alien defeats everything –
even Alien



How about a peek at the rest of the week's HW… ?

… you must mean *Pic* !