CS1 Lecture 41 Apr. 28, 2017

- Scores and grades so far (includes all but HW 7-10, disc. sec. 10, surveys score, and final exam)
  - Grades: 85% (130+ points) A, 70% (107+ points) B, 50% (77+ points) C, 40% (62+ points) D.
- HW 9 due Sunday 5pm.
- HW 10 available. Continues and substantially extends HW 9 Q2.

Today

1. Globals again
2. Q1: use configure to update widgets. Look in docs
3. Q2: for map type and pin, see google docs and modify get mapURL accordingly. Some widgets use tkinter IntVar objects
4. HW 10
   - Important first steps:
     - Get Twitter, Flickr accounts. Get credentials
     - Download oauth, httplib2, for authenticating with web services that require authorization (e.g. Twitter)
5. Working with Twitter
   - Authorizing
   - Using the twitter API from Python
   - Extracting information from JSON results returned by Twitter

1. Global variables, again

- Each function in a file has its own namespace. We call the variable names in a function's own namespace *local variables*
- There is another namespace for the file. Variable names in this namespace are called *global variables.*
- When Python processes a function definition, if it finds assignment statements (e.g. x = …), it creates local variables for the names on the l.h.s of the assignments *unless* there is a 'global …' statement in that function definition.

```
def init():
    global score
    score = 0
    root.mainloop()
```

```
def updateGame():
    global score
    …
    score = score + 1
    …
```

```
def updateGUI():
    …
    showScore(score)
    …
```

2. HW9 Q1 comments/hints

- To update widgets, use **configure**
  - in createGUI:

    statusLabel = Label(rootWindow, text="Play Nim")

  - Somewhere else (in updateGraphics or NimGame's take):

    statusLabel.configure(text= *your new text*)

  - Using only the code below in createGUI *does not* automatically keep score updated as gamesWon changes

    scoreLabel = tkinterLabel(…,text= *"You've won {}".formats(gamesWon)")*

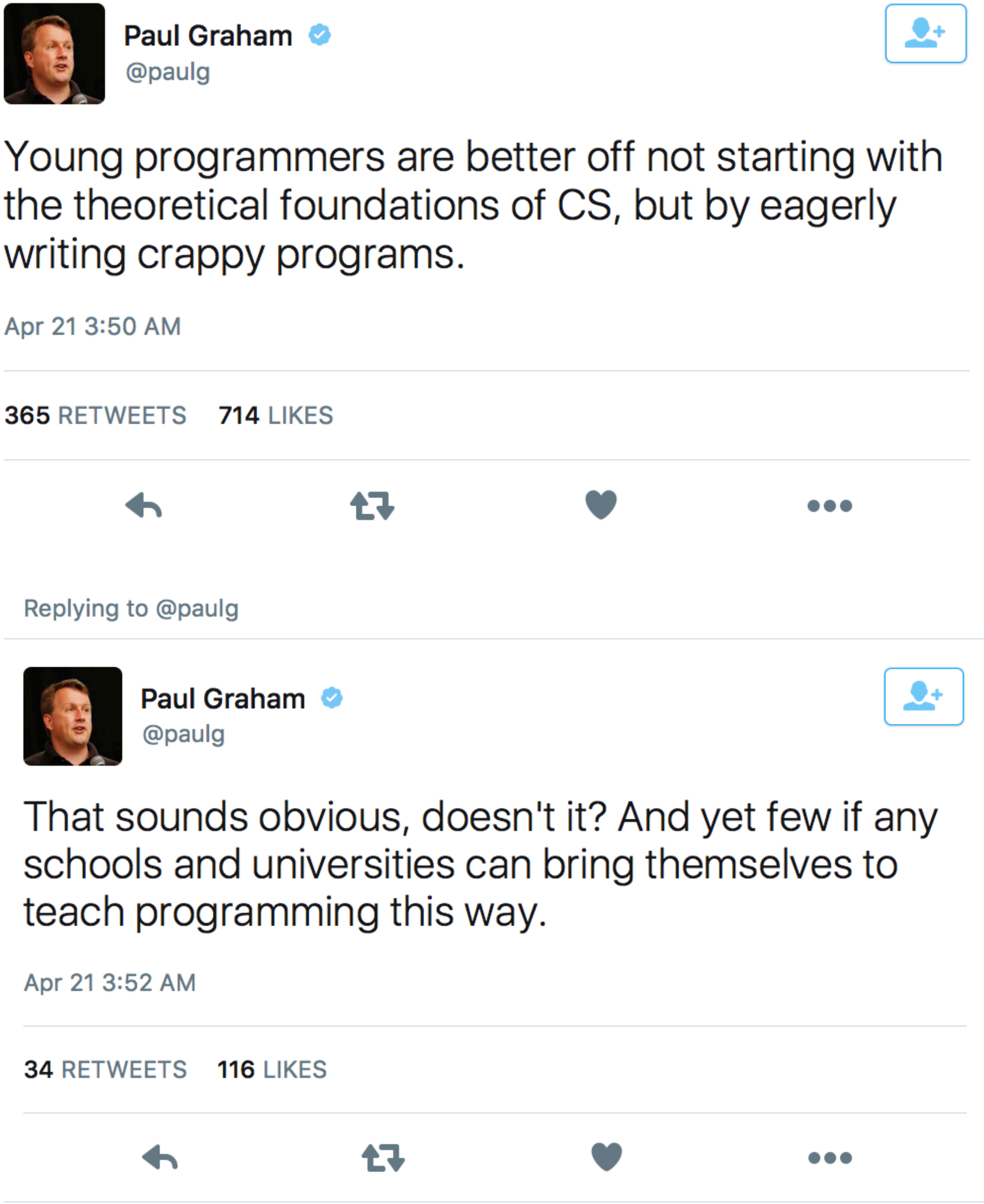
- To learn how to use new widgets or modify widget properties, Google: tkinter *widgetname*

3. HW9 Q2 To Do list

1. add Entry widget so you can change location
   - Upon button press, callback function (command) should read Entry, set mapLocation (global) variable, and call showMap()
2. enable zooming
   - Upon use of widget (button or whatever you choose), set zoomLevel (global) variable, call showMap()
3. enable changing map type
   - modify getMapURL to so that string representing request to Google Static maps API specifies desired map type (google: google static maps guide, and study API. Many examples in guide
   - add a new global variable for mapType. Modify retrieveMap to pass mapType to getMapURL.
   - Add widget(s) for map type. Upon use of widget, set mapType variable, and call showMap()
4. display pin at map center
   - modify getMapURL so that string representing request to Google Static maps API includes specification of pin at map center. Again see map docs.
5. Some tkinter widgets use special objects IntVar – containers for integers (think: a little box that holds a number). E.g. RadioButton is a nice widget to use for specifying mapType. See radioButton.py for simple example

HW 10 Philosophy?

- Paul Graham
  - Founder of Y Combinator, probably the most famous tech startup incubator
  - Creator of first ASP (web servies) provider



Paul Graham
@paulg

Young programmers are better off not starting with the theoretical foundations of CS, but by eagerly writing crappy programs.

Apr 21 3:50 AM

365 RETWEETS 714 LIKES

Replying to @paulg

Paul Graham
@paulg

That sounds obvious, doesn't it? And yet few if any schools and universities can bring themselves to teach programming this way.

Apr 21 3:52 AM

34 RETWEETS 116 LIKES

4. HW10

- Overview and demo
- Important first steps
  - Get Twitter account
  - Install httplib2, oauth2
- Start now!
  - More important than ever: do not write many lines of code before testing! This assignment has a lot of code and many little things can go wrong. If you add a lot of lines and then it crashes/doesn’t work, it can be very difficult to debug/find where the error is.
  - Add a few lines, test, add a few lines, test, …

5. Working with Twitter

demo: twitteraccess.py

The version provided in lecture notes will not work until you add your own credentials!

To "take apart" JSON tweet information returned by Twitter you need to study Twitter API: https://dev.twitter.com/rest/public/search

Next time

Details of working with Twitter:

- Specifying desired search location
- Extracting needed info from JSON-form results returned by Twitter:
  - extracting tweet text
  - extracting URLs from tweets
  - determining location of a tweet (so you can make a pin in that location on the map

Details of working with Flickr