Week 3 - Wednesday

COMP 1800

Last time

- What did we talk about last time?
- **`else`** and **`elif`** statements
- Visualization for Monte Carlo approximation
- Work time for assignments

Questions?

Strings

The string type

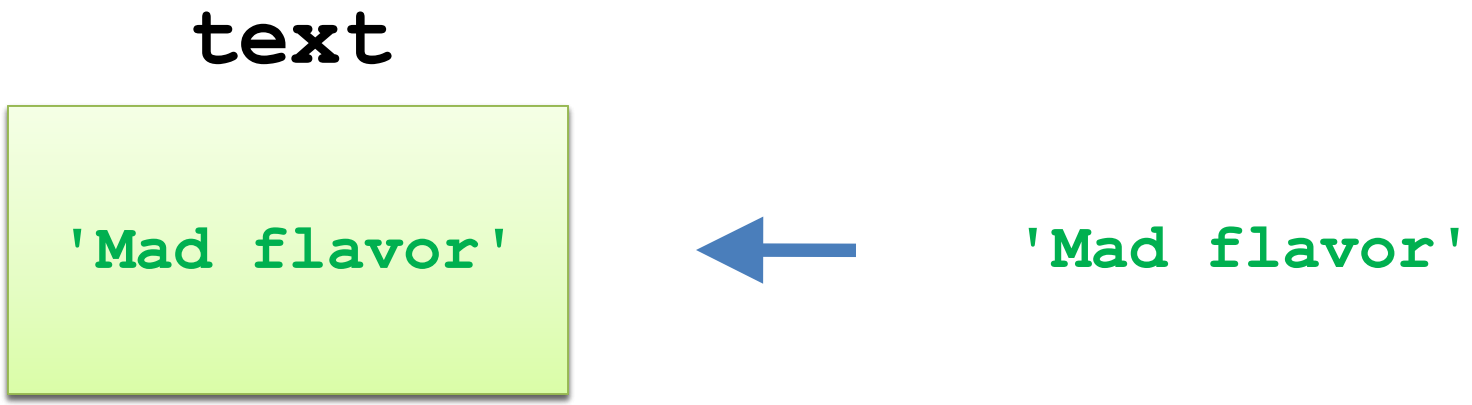
- The string type can hold any number of characters, not just a single letter
- A string literal is what we use whenever we print out text
- Strings can store text (up to some pretty large length, billions of characters on 32-bit Python and much more in 64-bit) from *most* of the different scripts in the world

```
text = 'message in a bottle'
```

Assignment for a string

```
text = 'Mad flavor'
```

- This line of code **stores** the string **`'Mad flavor'`** into the variable called **`text`**
- We must use either single quotes or double quotes so that Python knows we're talking about the text "Mad flavor" and not other variables or commands



Single vs. double quotes

- In Python, there's no difference at all between single quotes and double quotes

```
word1 = 'eggplant'
word2 = "eggplant" # exactly the same
```

- If the string contains single quotes, we'll usually use double quotes

```
message = "He earned an 'A'"
```

- Likewise, a string that contains double quotes is usually written with single quotes

```
sentence = 'Bob said, "I refuse."'
```

Operations on String Values

String operations

- You can use **+** to concatenate two strings together (to get a third string that is both of them stuck together)

```
place = 'boon' + 'docks'
print(place) # prints boondocks
```

- You can use ***** to get repetitions of a string

```
comment = 'yeah ' * 3
print(comment) # prints yeah yeah yeah
```

Concatenation with other types

- If you want to concatenate other types with strings, you have to convert them to strings first

```
number = 5
text = str(number) + ' fingers'
# text is '5 fingers'
```

- Note that concatenating a string doesn't change it
- You have to store the new string

```
word = 'figs'
word + word #'figsfigs' but doesn't store it
word = word + word #stores 'figsfigs'
```

String operations continued

- You can use the **`len()`** function to get the length of a string

```
author = 'Thomas Pynchon'
print( len(author) ) # prints 14
```

- You can use square brackets to get a particular character in the string
- Indexes start at 0
  - The first character in a string is at 0, the last is at its length - 1

```
movie = 'Dr. Strangelove'
print(movie[4]) # prints S
```

Indexing backwards

- You can also index from the back of a string instead of the front
- To do so, use negative numbers, where -1 is the last character in the string, -2 is the second to last, and so on

```
book = 'Harry Potter'
print(book[-1]) # prints r
print(book[-6]) # prints P
```

- Be careful! If you index past the end or the beginning of the string, your code will have an error

```
word = 'wombat'
print(word[6])  # error
print(word[-8]) # error
```

Slices

- If you want to get a substring (a part of a string) from a string, you can use the **slice** notation
  - Two numbers with a colon (**:**) in between
  - The first number is the starting point, the second number is the location after the ending point
  - If you subtract the first from the last, you'll get the length of the result

```
adjective = 'dysfunctional'
noun = adjective[3:6] # noun contains 'fun'
```

More on slices

- You can leave off the first index and Python will assume o

```
word = 'things'
width = word[:4] #width contains 'thin'
```

- Or you can leave off the last index and Python will assume the length of the string

```
intelligence = 'smart'
craft = intelligence[2:] #craft contains 'art'
```

The in operator

- The **`in`** operator will give a **`True`** if a string can be found inside another string and **`False`** otherwise
- This can be useful in **`if`** statements

```
animal = 'jellyfish'
if 'fish' in animal:
     print ("It's a fish!")
else:
     print ('No fish here.')
```

- You can also use **`not in`** if you want to see if a string is not inside another strong

Examples

- What are the possible indexes (both positive and negative) for the string **`'aardvark'`**?
- What slice is necessary to get the part of **`'bookkeeper'`** that contains **`'kk'`**?
- With an (unknown) string stored in a variable called **`phrase`**
  - Print out the middle letter if the length of **`phrase`** is odd
  - Print out the two middle letters if the length of **`phrase`** is even
- Write code that prints out **`'Yes!'`** if a string variable ends with the string **`'tion'`** and prints **`'No!'`** otherwise

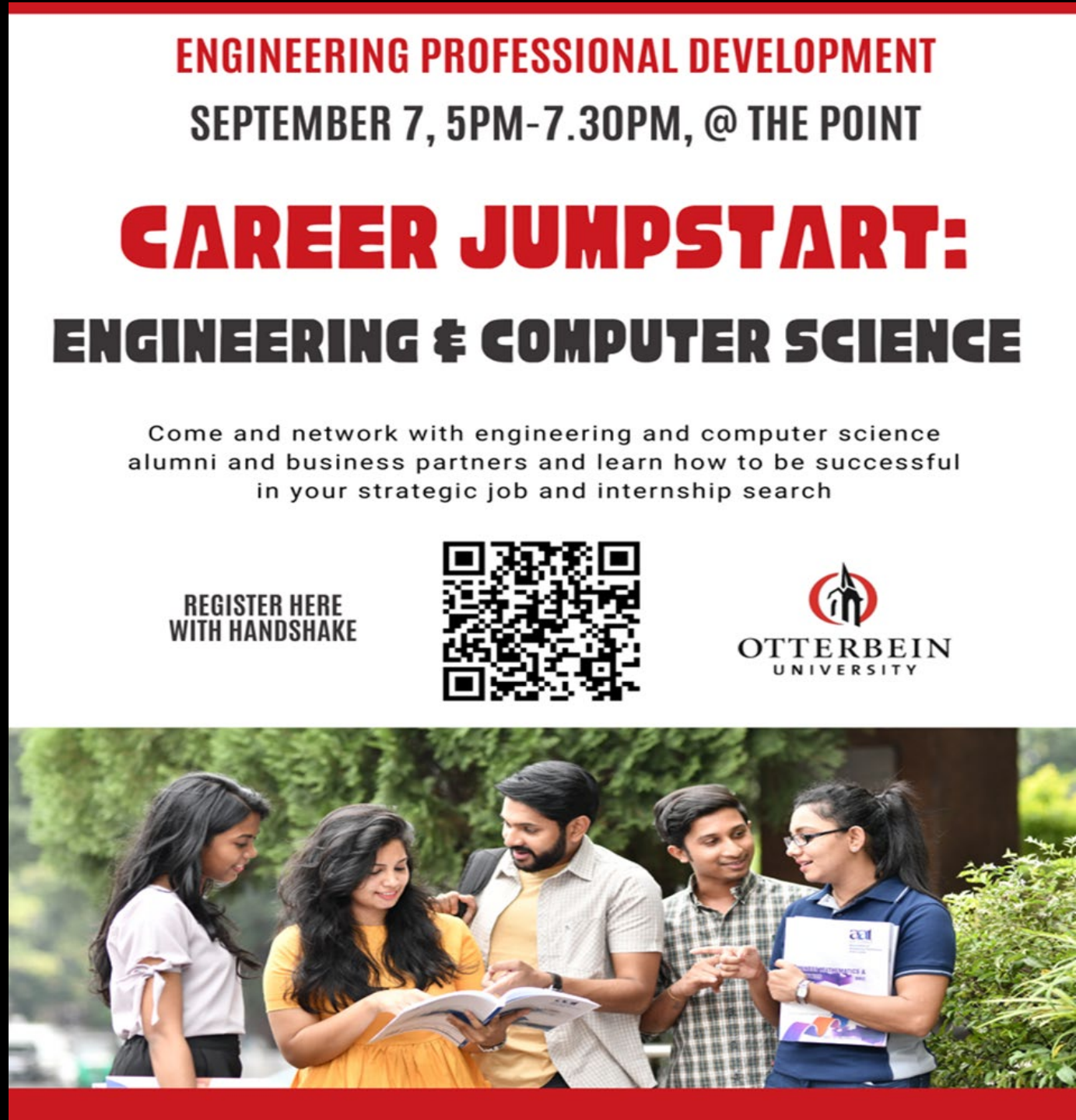
Quiz

Upcoming

Next time…

- Input
- Time to work on assignments

- 20 employers in the fields of Engineering and Computer Science
- 20 alumni members attending
- Free professional LinkedIn headshots
- Plenty of food and great conversations
- Build new connections on LinkedIn
- Door prizes
- Network with people in your field
- Learn about possible internships
- Gain new insights about your major
- Required event for all sophomores



Reminders

- Keep reading Section 3.2 of the textbook
- Work on Assignment 2
  - Due Friday by midnight
- Office hours from 4-5 p.m. **canceled** today for Faculty Assembly