Introduction to Computational Cognitive Science
AS.050.202 (Fall 2019)

Lab 1

Grusha Prasad and Najoung Kim

Before we jump in...

- Learning to code can be frustrating. You will likely spend hours debugging code only to realize you have a typo or a missing parenthesis. But you **will** get better with practice — at spotting bugs and fixing them with practice (though as Dumbledore said ...). So don't get too disheartened if it is difficult initially. And please ask questions!

- Google can be your best friend. If you don't know how to do something, or run into an error you don't understand there is probably a stackoverflow question about it. With practice you will also get better at knowing what to google.

Question 1

- Create a list called **courses** with the names AS.050.216, AS.050.202, AS.050.318, AS.050.102.
- Add the course AS.050.105 to this list.
- Rearrange the list so that the courses are ordered alphabetically.
- Print every item in the list on a separate line along with its index in the list.

Question 1: Possible Solution

```
courses = [“AS.050.216”, “AS.050.202”,
“AS.050.318”, “AS.050.102”]

courses.append(“AS.050.105”)

courses.sort()


for i, course in enumerate(courses):
   print(“%s %s”%(i, course))
```

Question 1: Possible Solution

```
courses = ["AS.050.216", "AS.050.202",
"AS.050.318", "AS.050.102"]

courses.append("AS.050.105")

courses.sort()

for i, course in enumerate(courses):
    print("%s %s"%(i, course))
```

sort() and append() alter the list in place. What happens if you had sorted_courses = courses.sort()?

What about:
sorted_courses = sorted(courses)

Question 1: Possible Solution

```
courses = [“AS.050.216”, “AS.050.202”,
“AS.050.318”, “AS.050.102”]

courses.append(“AS.050.105”)

courses.sort()

for i, course in enumerate(courses):
    print(“%s %s”%(i, course))
```

Remember the syntax!

Question 1: Possible Solution

```
courses = [“AS.050.216”, “AS.050.202”,
“AS.050.318”, “AS.050.102”]

courses.append(“AS.050.105”)

courses.sort()

for i, course in enumerate(courses):
    print(“%s %s”%(i, course))
```

Useful way to get both the item and the index of the item in a list.

Remember, items are zero-indexed

Question 1: Possible Solution

```
courses = ["AS.050.216", "AS.050.202",
"AS.050.318", "AS.050.102"]

courses.append("AS.050.105")

courses.sort()

for i, course in enumerate(courses):
    print("%s %s"%(i, course))
```

Convenient way to include variables in strings.
You can also use "{} {}".format(i, course)

Question 2

- Create a string called **coursename** with the title of this class.
- Create a copy of **coursename** called **str_copy**.
- Delete the word “Introduction” from **str_copy.**
- Create a list called **chars** which is a list of all the characters in **coursename.**
- Create a list called **words** which is a list of all the words in **coursename.**
- Create a copy of **words** called **words_copy** and delete the word “Introduction” from the list.
- Print **words**, **word_copy, coursename and str_copy.**

Question 2: Possible Solution

```
coursename = “Introduction to Computational
Cognitive Science”

str_copy = coursename
str_copy = str_copy.replace(“Introduction”, “”)

chars = list(coursename)

words = coursename.split(“ “)
words_copy = words
words_copy.pop(0)
print(words, words_copy, coursename, str_copy)
```

Question 2: Possible Solution

```
coursename = "Introduction to Computational
Cognitive Science"

str_copy = coursename
str_copy = str_copy.replace("Introduction", "")

chars = list(coursename)

words = coursename.split(" ")
words_copy = words
words_copy.pop(0)
print(words, words_copy, coursename, str_copy)
```

replace() unlike with sort() does not work "in place". You need to assign its output to a new (or same) variable.

Question 2: Possible Solution

```
coursename = “Introduction to Computational
Cognitive Science”

str_copy = coursename

str_copy = str_copy.replace(“Introduction”, “”)

chars = list(coursename)

words = coursename.split(“ ”)

words_copy = words

words_copy.pop(0)

print(words, words_copy, coursename, str_copy)
```

You can convert one type to another. For example also, int(“3”).

Question 2: Possible Solution

```
coursename = “Introduction to Computational
Cognitive Science”

str_copy = coursename
str_copy = str_copy.replace(“Introduction”, “”)

chars = list(coursename)

words = coursename.split(“ “)
words_copy = words
words_copy.pop(0)
print(words, words_copy, coursename, str_copy)
```

Can replace “ “ with any other separator.

Question 2: Possible Solution

```
coursename = “Introduction to Computational
Cognitive Science”

str_copy = coursename

str_copy = str_copy.replace(“Introduction”, “”)

chars = list(coursename)

words = coursename.split(“ “)

words_copy = words

words_copy.pop(0)

print(words, words_copy, coursename, str_copy)
```

Remember zero-indexing! What would words_copy.pop(3) delete? Also pop() is inplace like sort()

Question 2: Possible Solution

```
coursename = "Introduction to Computational
Cognitive Science"

str_copy = coursename

str_copy = str_copy.replace("Introduction", "")

chars = list(coursename)

words = coursename.split(" ")

words_copy = words

words_copy.pop(0)

print(words, words_copy, coursename, str_copy)
```

coursename != str_copy
BUT words == words_copy

If you don't want to alter the original list, you need to create a **deepcopy**

Spot the error(s)!

Take in a list of numbers. If the result of dividing the third number by the fourth number greater than 2.5 then return a list where the third number is 2.5. Otherwise return the same list.

```
def thirdnum(l):
    x = l[3]/l[4]
      if x > 2.5:
         X[3] == 2.5
      else:
         return(l)
```

Spot the error(s)!

Take in a list of numbers. If the result of dividing the third number by the fourth number greater than 2.5 then return a list where the third number is 2.5. Otherwise return the same list.

```
def thirdnum(l):
   x = l[3]/l[4]
    if x > 2.5:
      X[3] == 2.5
    else:
      return(l)
```

Inconsistent indentation. Also indenting with tab is not the same as indenting with spaces!

Spot the error(s)!

Take in a list of numbers. If the result of dividing the third number by the fourth number greater than 2.5 then return a list where the third number is 2.5. Otherwise return the same list.

```
def thirdnum(l):
   x = l[3]/l[4]
    if x > 2.5:
      x[3] == 2.5
    else:
      return(l)
```

Typo! "Variable referenced before assignment"

Spot the error(s)!

Take in a list of numbers. If the result of dividing the third number by the fourth number greater than 2.5 then return a list where the third number is 2.5. Otherwise return the same list.

```
def thirdnum(l):
    x = l[3]/l[4]
     if x > 2.5:
       X[3] == 2.5
     else:
       return(l)
```

Only single equals (=). Double equals is for assessing truth value of the statement.

Spot the error(s)!

Take in a list of numbers. If the result of dividing the third number by the fourth number greater than 2.5 then return a list where the third number is 2.5. Otherwise return the same list.

```
def thirdnum(l):
  x = l[3]/l[4]
    if x > 2.5:
      X[3] == 2.5
    else:
      return(l)
```

This function will only return a list if the value is not greater than 2.5. It will return nothing otherwise. Either delete “else” or add a return statement within “if”.

Spot the error(s)!

Take in a list of numbers. If the result of dividing the third number by the fourth number is greater than 2.5 then return a list where the third number is 2.5. Otherwise return the same list.

```
def thirdnum(l):
    x = l[3]/l[4]
     if x > 2.5:
       X[3] == 2.5
     else:
       return(l)
```

In Python 2, if the list had integers, it would round down for integer division. This would result in the wrong answer sometimes.

This kind of bug is the most "dangerous" because you might not even realize you have it unless you have the right test case.

Standing on the shoulders of giants…

- Useful list of commonly made mistakes: https://pythonforbiologists.com/29-common-beginner-errors-on-one-page

- Class list of bugs that annoyed you and tricks you found helpful (so we can learn from each other!): https://docs.google.com/document/d/1uLKpKZ4IGT2n0NyEL0zenrt7JrjYWLqlHMIu-gHezx0/edit?usp=sharing