Nested Loops with Lists

COSC 101: Intro to Computing I
October 11, 2017
Exam 1

• View your grades and feedback on Gradescope
• Pickup your paper exam in McGregor 209A
• Exam Solution Dinner: Tomorrow 6:00–8:00pm
  RSVP at shoutkey.com/lead
Exercise #1: What are some functions that can be performed on lists?
Exercise #2: What is the output of the following code? If there are any errors, correct them.

```python
L1 = [[1], [2], [3], [4], [5]]
for i in range(6, 11):
    L1.append([i])
print(L1)
```

```
[ [1], [2], [3], [4], [5], [6], [7], [8], [9], [10] ]
```
Exercise #3: What is the output of the following code? If there are any errors, correct them.

```python
L1 = [[1], [2], [3], [4], [5]]
for itm in L1:
    for i in range(itm[-1], 7):
        L1.append([i])
print(L1)
```

```python
[[1, [1], [2], [3], [4], [5], [6]], [2, [2], [3], [4], [5], [6]], [3, [3], [4], [5], [6]], [4, [4], [5], [6]], [5, [5], [6]]
]
```
How could we modify the code from Exercise #3 to produce the following output?

```
[ [1, 2, 3, 4, 5, 6 ],
  [2, 3, 4, 5, 6 ],
  [3, 4, 5, 6 ],
  [4, 5, 6 ],
  [5, 6 ]
]
```

```
L1 = [[1], [2], [3], [4], [5]]
for itm in L1:
    for i in range(itm[-1]+1, 7):
        L1.append(i)
print(L1)
```
Exercise #4: What is the output of the following code? If there are any errors, correct them.

L1 = []
L2 = [1, 2, 3]
L3 = [4, 5, 6]

for i in L3:
    for j in L2:
        L1.append(L2.pop())

print("L1 = ", L1)
print("L2 = ", L2)
print("L3 = ", L3)

L1 = [3, 2, 1]
L2 = []
L3 = [4, 5, 6]
**Exercise #5:** What is the output of the following code? If there are any errors, correct them.

```python
def print_list(a_list):
    for itm in a_list:
        print(itm, end="", )
print()  # This print() is unnecessary

def print_lists(a_list):
    for itms in a_list:
        for itm in itms:
            print(itm, end="", )
        print()  # This print() is unnecessary

list1 = [1, 2, 3, 4]
print(list1)
print_list(list1)
print_lists(list1)

list2 = [[1, 2, 3], [2, 3, 4]]
print(list2)
print_list(list2)
```
Exercise #5: What is the output of the following code? If there are any errors, correct them.

```
[1, 2, 3, 4]
1, 2, 3, 4,
TypeError: 'int' object is not iterable
[[[1, 2, 3], [2, 3, 4]]
[1, 2, 3], [2, 3, 4],
1, 2, 3,
2, 3, 4,
```
How can you modify the functions from Exercise #5 to only print commas between list elements?

```python
def print_list2(a_list):
    for i in range(len(a_list)):
        if i < len(a_list) - 1:
            print(a_list[i], end="", )
        else:
            print(a_list[i])

def print_lists2(a_list):
    for its in a_list:
        for i in range(len(itm)):
            if i < len(itm) - 1:
                print(itm[i], end="", )
            else:
                print(itm[i])
```
Exercise #6: Write a function `count_distinct` that takes a list and returns the number of distinct items.

```python
def count_distinct(L):
    ''' (list of object) > int
    Returns the number of distinct objects in L. Expects L to contain at least one item.
    >>> count_distinct([10, 20, 10, 30, 20])
    3
    >>> count_distinct([10, 20, 10, 30, 40])
    4
    '''
```
Exercise #6: Write a function `count_distinct` that takes a list and returns the number of distinct items.

```python
def count_distinct(L):
    L = L[:]  # make copy to avoid modifying L.sort()
    current = L[0]
    distinct = 1
    for i in range(1, len(L)):
        if L[i] != current:
            current = L[i]
            distinct += 1
    return distinct
```
Exercise #7:

```python
def word_to_lists(word):
    ''' (string) > list<list<str>>
    Returns a list with sublists for each character in the word. The contain length of the sublists is the index of the element in the outer list. The items in the sublist are the characters in word in order.
    >>> word_to_lists('cat')
    [[], ['c'], ['c', 'a']]
    '''
```
Exercise #7:

```python
def word_to_lists(word):
    L = []
    for i in range(len(word)):
        L.append([])
        for j in range(i):
            L[i].append(word[j])
    return L
```