

Advanced Exercises with Loops

CS 8: Introduction to Computer Science, Winter 2019
Lecture #9

Ziad Matni, Ph.D.
Dept. of Computer Science, UCSB

SHORT Online Class Evaluation

- You will get an email invitation. Please go to the URL in there and fill out a mid-quarter evaluation.
- This is strictly about Prof. Matni and the class. It is **ANONYMOUS**
 - I promise to take your feedback seriously, so leave some constructive feedback too! ☺
- Open now and **will close tomorrow** just before midnight!

If I get at least 85% participation, **EVERYONE** gets
+2 pts on the midterm #1 exam score!

Administrative

- Please note that next Monday (2/18) is a Uni. Holiday
- Hw05 – due next week Wednesday (2/20)
- Lab05 starts tomorrow (is due Monday 2/18)
- Midterm Exam #1 grades will be posted on Wednesday
- Online class evaluation due on Tuesday by midnight

Lecture Outline

- Loops with Turtle
- Nested Loops
- Accumulated Loops

More Drawing Abstraction

- *Drawing any regular polygon using Turtle and loops!*

```
def drawPolygon(myTurtle, sideLength, numSides):  
    turnAngle = 360 / numSides  
    for i in range(numSides):  
        myTurtle.forward(sideLength)  
        myTurtle.right(turnAngle)
```

Simpler Drawing By Repetition

- *Drawing a spiral using Turtle and loops!*

```
def drawSpiral(myTurtle, maxSide):  
    for sideLength in range(1, maxSide+1, 5):  
        myTurtle.forward(sideLength)  
        myTurtle.right(90)
```

Nested Loops

- What would this do?

```
listX = [ [1, 2, 3], [4, 5, 6], [7, 8, 9] ]  
for i in listX:  
    for j in i:  
        print(j)
```

Exercises with Nested Loops

```
def drawRectangle(width, height):
    """ print a rectangle with given width
        and height using the character *
        (instead of turtle)
```

For example `drawRectangle(5,3)`
should print

"""

Exercises with Nested Loops

```
def drawRectangle(width, height):  
    for w in range(width):  
        for h in range(height):  
            print("*")  
    print("")
```

PLEASE NOTE THE INDENTATIONS!!!!

Applying while

Example of an Accumulated Sum

Unlimited data entry (please note corrections from last week's slides)

```
AllGrades = 0                      # (1) initialize for accumulated sum
grade = input("enter grade or q to quit: ")
while grade != "q":                # (2) check condition
    AllGrades = AllGrades + int(grade)      # accumulate sum
    grade = input("enter grade or q to quit: ")  # ask again

# While loop has ended (no indents after here),
# now you can do other stuff...
print("Total grades is:", AllGrades)
print("You're all done now!")
```

Let's try it out!

Exercises with Accumulation 1

- Useful for "accumulating" something while going through a collection.
- Finish this function:

```
def countElements(MyL):  
    """ returns the number of elements in list MyL """
```

Exercises with Accumulation 1

- Useful for "accumulating" something while going through a collection.
- Finish this function:

```
def countElements(MyL):  
    """ returns the number of elements in list MyL """  
    sum = 0  
    for item in MyL:  
        sum = sum + 1  
    return sum
```

Accumulation Operators

- Instead of:

$X = X + 6$

Or

$Y = Y * 9$

Or

$Z = Z / 5$

- You can do:

$X += 6$

Or

$Y *= 9$

Or

$Z /= 5$

Exercises with Accumulation 2

- Useful for "accumulating" something while going through a collection.
- Finish this function:

```
def countOddNumbers(lst):  
    """ returns the number of odd numbers in lst """
```

Exercises with Accumulation 2

- Useful for "accumulating" something while going through a collection.
- Finish this function:

```
def countOddNumbers(lst):  
    """ returns the number of odd numbers in lst """  
    oddItems = 0  
    for item in MyL:  
        if item % 2 == 1:  
            oddItems += 1  
    return oddItems
```

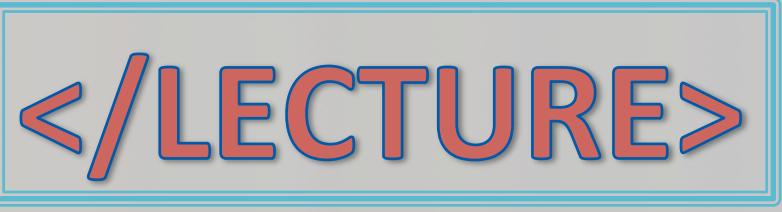
Exercises with Accumulation 3

- Useful for "accumulating" something while going through a collection.
- Finish this function:

```
def countWords(sentence):  
    """ returns the number of words in the string sentence """
```

YOUR TO-DOS

- Start **Lab5** (lab tomorrow ; turn it in by **Monday, 2/18**)
- Start **HW5** (due on **Wednesday, 2/20**)
- Make a playlist of songs that make you feel good



</LECTURE>