

# Loops

CS 8: Introduction to Computer Science, Spring 2019  
Lecture #7

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

# Administrative

---

- **Hw03 due today!**
- Hw04 – due next week
- Lab02 – due on Sunday by midnight (11:59 pm) on  
**Gradescope!**
- **Midterm Exam #1 is NEXT WEEK on Thu., May 2<sup>nd</sup>**
  - I'll put up sample problems after Thu. this week

# Lecture Outline

---

- Loops using **for** and **while**

# print() options

---

- `print()` has certain defaults:
  - If you separate items with a comma (,) you get a space in between
  - Your item get a newline at the end
- To over-ride these defaults use:
  - `sep=""` to create your own separator
  - `end=""` to create your own ending
- For example, try these on IDLE:
  - `print("hello", "honey")` vs. `print("hello", "honey", sep="!")`  
vs. `print("hello", "honey", end="...")`

# Class Exercise

---

Get together with 2 or 3 other people around you and answer this question.

You can use your notes from last time. You can use your computers:

- a) Write a short Python code that asks a user their age. Once you do that, decide whether to print out “**Your age is an even number!**” or “**Your age is an odd number!**” depending on their answer.
  
- b) Now modify your code so that it can detect if someone entered a number less than 1 as their age. If so, print out a rejection message (“**BAD AGE!**”) and quit.  
Challenge: do this twice: once by using the **and** operator and once *without* using **and** (using nested-if statements)

# Class Exercise

---

```
age = int(input("How old are you? "))

if (age % 2 == 0):
    print("Your age is an even number!")

else:
    print("Your age is an odd number!")
```

# Class Exercise

---

```
age = int(input("How old are you? "))

if (age % 2 == 0) and (age > 0):
    print("Your age is an even number!")
elif (age % 2 != 0) and (age > 0):
    print("Your age is an odd number!")
else:
    print("You have entered an illegal age!")
```

# Class Exercise

---

```
age = int(input("How old are you? "))

if (age > 0):
    if (age % 2 == 0):
        print("Your age is an even number!")
    else:
        print("Your age is an odd number!")

else:
    print("You have entered an illegal age!")
```

# for Loops

```
for x in range(7):  
    print (x)
```

WHAT DO YOU THINK THESE  
LOOPS PRINT OUT?

```
for y in range(2, 9):  
    print (y - 2)
```

```
for item in range(5, -1, -1):  
    if item == 0:  
        print(item, "Blast off!!")  
    else:  
        print(item)
```

# Repetition with a **while** loop

---

**while** *condition*:

*# executes over and over until a condition is False*

- Used for **indefinite iteration**
  - When it isn't possible to predict how many times a loop needs to execute, unlike with **for loops**
- We use **for** loops for **definite iteration**  
(e.g., the loop executes exactly **n** times)

# Repetition with a **while** loop

---

**while** *condition*:

*# executes over and over until a condition is False*

- While loops **won't run at all** if *condition* starts out as false
- While loops **run forever** if *condition* never becomes false  
(i.e. if it always stays true)
- So care must be done in designing these sort of loops.

# Applying while

Can be used for counter-controlled loops:

```
n = 500
counter = 0                      # (1) initialize
while counter < n:                # (2) check condition
    print(counter * counter)
    counter += 1                  # (3) change state
# also, note that counter += 1 is equivalent to counter = counter + 1
```

- But NOTE that this is a **definite** loop example – it's easier to use a **for** loop:

```
for counter in range (500):      ...etc...
```

# Applying while

This is a better application example – unlimited data entry:

```
AllGrades = 0                                # (1) initialize
grade = input("enter grade or q to quit: ")
while grade != "q":                          # (2) check condition
    AllGrades += int(grade)                  # (3) process grade
    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!")
```

# YOUR TO-DOS

---

- Finish reading **Chapter 5**
- Finish **HW4** (due **TUESDAY**)
- Finish **Lab2** (turn it in by **Sunday**)
  
- Whistle while you work

</LECTURE>