CSC 111: Intro to Computer Science through Programming

Spring 2017
Prof. Sara Mathieson
Admin

- Homework 5 due Tues after spring break
- Lab this week will be like TA hours (optional time to ask questions and work on homework/studying)
- Pick up your practice midterm if you haven’t done so already
- Pick up a midterm study sheet if you haven’t done so already
- Office hours **TODAY 3-5pm** in Ford 355
Midterm logistics recap

- In the Nielson library starting Tuesday morning
- Time limit: 2 hours
- No phones, computers, book, notes, etc
- No discussion of the exam with anyone else
- You may use a cheat sheet (double sided), don’t have to turn in
- Please do not write on other sheets (backs of exam okay)
- For code: do not worry about comments, variable names, perfect syntax
Outline: 3/6

+ Lab 3 examples
+ Finish midterm review (emphasis on practice with Booleans)
+ While loops
+ Wednesday: start Graphics!
Lab 3 Examples
(selected by Aditya)
Lab 3 Feedback

1) Reassign the count to itself (unnecessary)

2) Angle passed to `math.sin(angle)` should be in radians

3) `random.randint(0, 3)` to go over ['A', 'C', 'G', 'T']; use `len()` to make it more generic

4) In Part E, make sure that the unknown case is first
# CSC 111, Lab 3, Part E
# Author: Hening Zheng and Maggie Carttar

# This program counts how many bases are the same, how many are different, and how many are unknown.

def main():
    human = "ATA?CAAGACCTCGTTATTAATACGGGCGCATGTAATCCTATC?GA"
    chimp = "ATAACAAGAGCTAGTTATATACTGCGGATGAGAAATCCTATAGGA"

    # Variables all start at zero
    same = 0
    different = 0
    unknown = 0

    # Calculate how many bases are unknown, same, different
    for i in range(len(human)):
        # Unknown goes up by 1 if either or both human and chimp bases are "?"
        if human[i] == "?" or chimp[i] == "?":
            unknown = unknown + 1

        # Same goes up by 1 if human and chimp bases are the same
        elif human[i] == chimp[i]:
            same = same + 1

        # Different goes up by one if human and chimp bases are different
        else:
            different = different + 1

    print(same, "bases are the same.")
    print(different, "bases are different.")
    print(unknown, "bases: either human or chimp or both unknown.")

main()
```python
# CSC 111, lab 3 E
# Authors: Deniz Keles and Blanca Martin

def main():
    human = "ATA?CAAGACCTCGTTATTAATACGGCGCCATGTAGTAATCCTATC?GA"
    chimp = "ATAACAAGAGCTAGTTATTA?TACTGCCATGTGAGAAATCCTATAGGA"

    count_same = 0
    count_different = 0
    count_unknown = 0

    # for the length of the DNA sequence,
    # the loop compares the bases in human and chimps
    for i in range(len(human)):
        # If human/chimp or both are unknown
        if human[i] == '?' or chimp[i] == '?':
            count_unknown = count_unknown + 1

        # Looks for when the bases are different
        elif human[i] != chimp[i]:
            count_different = count_different + 1

        # Looks for when the bases are the same
        else:
            count_same = count_same + 1

    print(str(count_same), 'bases are the same."
    print(str(count_different), 'bases are different."
    print(str(count_unknown), 'bases: either human or chimp or both unknown."

main()
```
Midterm review
Practice midterm feedback

Part A (triangle name)

1) Cannot subtract strings the way we can add them using concatenate

2) Some people used one for loop with an if/else condition – this does work, but becomes more complicated and is harder to get right

3) Lots of hardcoding of the number (5, 4, etc) instead of using the variable number
Practice midterm feedback

Part A (triangle name)

1) Cannot subtract strings the way we can add them using concatenate

2) Some people used one for loop with an if/else condition – this does work, but becomes more complicated and is harder to get right

3) Lots of hardcoding of the number (5, 4, etc) instead of using the variable number

Part B (palindrome)

1) Using new_word instead of calling reverse(word)

2) Indexing vs. going over the characters when using a for loop

3) == vs. =
def main():
    string = input("Enter a string: ")

    if palindrome(string):
        print("\n>> " + string, "is a palindrome\n")
        for i in range(len(string)):
            if i == 0 or i == len(string)-1:
                print(string)
            else:
                stars = "*" * (len(string)-2)
                print(string[i] + stars + string[i])
    else:
        print("\n>> " + string, "is not a palindrome\n")
        print(string + reverse(string))

Enter a string: tacocat
>> tacocat is a palindrome

Enter a string: hello
>> hello is not a palindrome
helloolleh
Practice Midterm Exercise: Booleans

TODO: write a function to determine whether or not a string has a repeated letter

```python
>>> repeated_letter("hello")
True
>>> repeated_letter("goodbye")
True
>>> repeated_letter("spring")
False
>>> repeated_letter("boolean")
True
>>> repeated_letter("fall")
True
>>> repeated_letter("break")
False
```
Slicing recap: Lab 4

# selected tweets from President McCartney

tweet1 = "Great to see Northampton's growth and
   @MayorNarkewicz's leadership recognized!
   #ProudNeighbors"

tweet2 = "SGA, future leaders of the world,
   working hard for Smithies today
   @smithcollege #SmithiesLead"

tweet3 = "Gluten-free birthday lunch at Dawes.
   Thanks to Lisa and Raj, chefs extraordinaire
   @smithcollege #Yummy #GF"

tweet4 = "Northamptonites and Smith Notables
   sing Halo together to warm up at Silver
   Chord Bowl #HampArts"

tweet5 = "Shout out to January '17 grads
   @smithcollege #WomenForTheWorld
   #TheyAreReady #SmithiesLead"

Goal output:

>>> ProudNeighbors SmithiesLead Yummy GF HampArts WomenForTheWorld TheyAreReady SmithiesLead >>>