CompSci 101
Introduction to Computer Science

February 23, 2016
Prof. Rodger
from xkcd

Drinking Fountains

I avoid drinking fountains outside bathrooms because I'm afraid of getting trapped in a loop.
Announcements

• Reading and RQ due next time
• APT 4 out today, due in a week
• Do not discuss exam1 with anyone until it is handed back

• Today:
  – Loops – While, While True
  – Problem Solving
Is number a Prime number?
Bit.ly/101sp16-0223-A

def isPrime(number):
    if number<4:
        return True
    for n in range(4,number):
        if number/n * n == number:
            return False
    return True
Lab this week

• Working with While loops

• Demo: one-dimensional Walker
While loops

• Repetition when you stop a loop based on a condition

• while CONDITION:
  BODY

  – As long as condition is true, keep executing loop.
  – Must have an update in the body to get closer to condition being false
Examples for while

• Playing chess
  while (game not over)
  play game
  (game must get closer to ending)

• Finding the 100\textsuperscript{th} prime
Mystery While example

bit.ly/101sp16-0223-0

def mystery(strng):
    count = 0
    result = ""
    while count < 5:
        result += strng[count] + strng[count]
        count += 1
    result += strng[count:]
    return result

print mystery("September")
Problem: Given a number, want the largest list of unique digits from 1 to x whose sum is less than or equal to the number

- Given 5
  Answer is 1 + 2
- Given 6
  Answer is 1+2+3
bit.ly/101sp16-0223-1

def addDigitsTilSum2(total):
    sum = 0
    digs = []
    for n in range(1, 10):
        sum += n
        if sum > total:
            break
        digs.append(n)
    return digs
Looping with while
– not sure when to stop

• Playing chess
• Determining the $100^{th}$ prime number

• Another way – while True
  – Must have ways to break out of infinite loop
  – Must have update – gets closer to ending
while condition vs while True

while condition:
  body
continue

while True:
  body
  if condition:
    break
  continue

While condition is true - must update
  - must get closer to making condition false
  - use break to exit
While True

initialize

while True:
    if something:
        break
    if something2:
        update

update

Continue or return
def addDigitsTilSum(total):
    sum = 0
    num = 1
    digs = []
    while(True):
        sum += num
        if sum > total:
            break
        digs.append(num)
        num += 1
    return digs
Georges Harik and Noam Shazeer created the underlying data that led to AdSense.

Harik and Shazeer spent years analyzing data on webpages, trying to understand clusters of words and how they worked together. The data they gather wound up being used by Google for its AdSense product, which analyzed webpages for words, and then stuck ads on them.
Problem:
Find the first duplicate adjacent words in a phrase

• “Did I make make a mistake in in this?”

• Convert to list

• Answer is 2nd word (start counting at 0)
def positionDuplicate(phrase):
    words = phrase.split()
    if len(words)< 2:
        return -1
    pos = 0
    while True:
        pos = pos+1
        return pos
APTs solved in a similar way

- NameGroup
- Pikachu