SCC461 – Programming for Data Scientists

Leandro Marcolino

Lancaster University

Week 9

Outline

What will be printed?

```
def callMe(input):
        print(input);
        if (input >= 0):
                callMe(input - 1);
        return 0;

x = callMe(5);
print(x);
```

What will be printed?

```
a = [1, 2, 3, 4, 5];

b = a;

b[1] = 999;

print(a);
```

What will be printed?

```
def callMe(input):
    input = 1;
    return;

myInput = 0;
callMe(myInput);
print(myInput);
```

What will be printed?

```
def callMe(input):
    input[0] = 1;
    return;

myList = [0];
callMe(myList);
print(myList[0]);
```

What will be printed?

```
myList = [3, 4, 7, 8]

for x in myList:
    print(x)
```

What will be printed?

```
myList = [3, 4, 7, 8]

for x in range(len(myList)):
    print(x)
```

What will be printed?

```
myList = [3, 4, 7, 8]

for x in range(len(myList)):
    print(myList[x])
```

What will be printed?

```
myList = [3, 4, 7, 8]

for x in myList:
    x = 5

print(myList[0])
```

What will be printed?

```
myList = [3, 4, 99, 95, 7]
x = -1
xPos = -1

for y in range(len(myList)):
    if (myList[y] > x):
        x = myList[y]
        xPos = y

print(x)
print(xPos)
```

What will be printed?

```
myList = [3, 4, 99, 95, 7]
x = -1
xPos = -1

for y in range(len(myList)):
    if (y == 2):
        continue
    if (myList[y] > x):
        x = myList[y]
        xPos = y

print(x)
print(xPos)
```

What will be printed?

```
myList = [3, 4, 99, 95, 7]
x = -1
xPos = -1

for y in range(len(myList)):
    if (y == 2):
        break
    if (myList[y] > x):
        x = myList[y]
        xPos = y

print(x)
print(xPos)
```

What will be printed?

```
myList = [3, 4, 7, 10, 23, 2, 3]
specialList = []

for x in myList:
    if (x % 2 == 0):
        specialList.append(x)

print(specialList[1])
```

What will be printed?

```
myList = [3, 4, 7, 10, 23, 2, 3]
specialList = []

for x in myList:
    if (x % 2 != 0):
        specialList.append(x)

print(specialList[1])
```

What will be printed?

```
class Point:

    def __init__(self, x=0, y=0):
        self.x = x
        self.y = y

p = Point()
q = Point(3,4)

print(p.x)
print(q.x)
```

What will be printed?

```
class Point:

    def __init__(self, x=0, y=0):
        self.x = x
        self.y = y

p = Point()
q = Point(3,4)

p = q

q.x = 99

print(p.x)
print(q.x)
```

What will be printed?

```
class Point:
    def __init__(self, x=0, y=0):
        self.x = x
        self.y = y

    def distance_from_origin(self):
        return ((self.x ** 2) + (self.y ** 2))
            ** 0.5

def distance_from_origin():
    return -99

p = Point(1,1)
print(p.distance_from_origin())
print(distance_from_origin())
```

What will be printed?

```
def x(pt):
    pt.x = -1;
    print(pt.x);

pt1 = Point();
pt2 = Point();

pt1.x = 5;
pt1 = pt2;

x(pt2);
print(pt1.x);
```

What will be printed?

```
def print_list(node):
    while node is not None:
        print(node.content)
        node = node.next

node1 = Node(5)
node2 = Node(99)
node3 = Node(-3)
node4 = Node(67)

node1.next = node4
node3.next = node2
node4.next = node3

print_list(node1)
```

What will be printed?

```
s1 = Stack();
s2 = Stack();

s1.push(0)
s1.push(1)
s1.push(2)

for x in range(3):
    s2.push(s1.pop())

for x in range(3):
    print(s2.pop())
```

What will be printed?

```
q1 = Queue()
q2 = Queue()

q1.insert(0)
q1.insert(1)
q1.insert(2)

for x in range(3):
    q2.insert(q1.remove())

for x in range(3):
    print(q2.remove())
```

What will be printed

```
s1 = Stack()
q1 = Queue()

s1.push(0)
s1.push(1)
s1.push(2)

for x in range(3):
    q1.insert(s1.pop())

for x in range(3):
    print(q1.remove())
```

CW 7

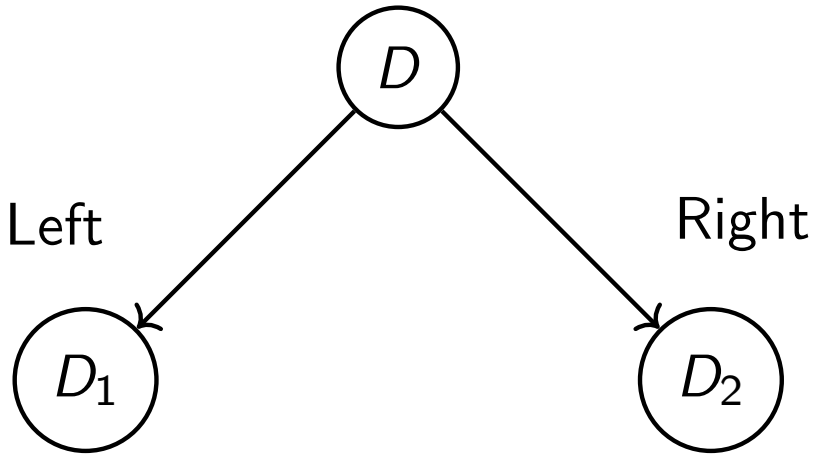
- File FibonacciQueue.py
- File PriorityQueue.py

Trees

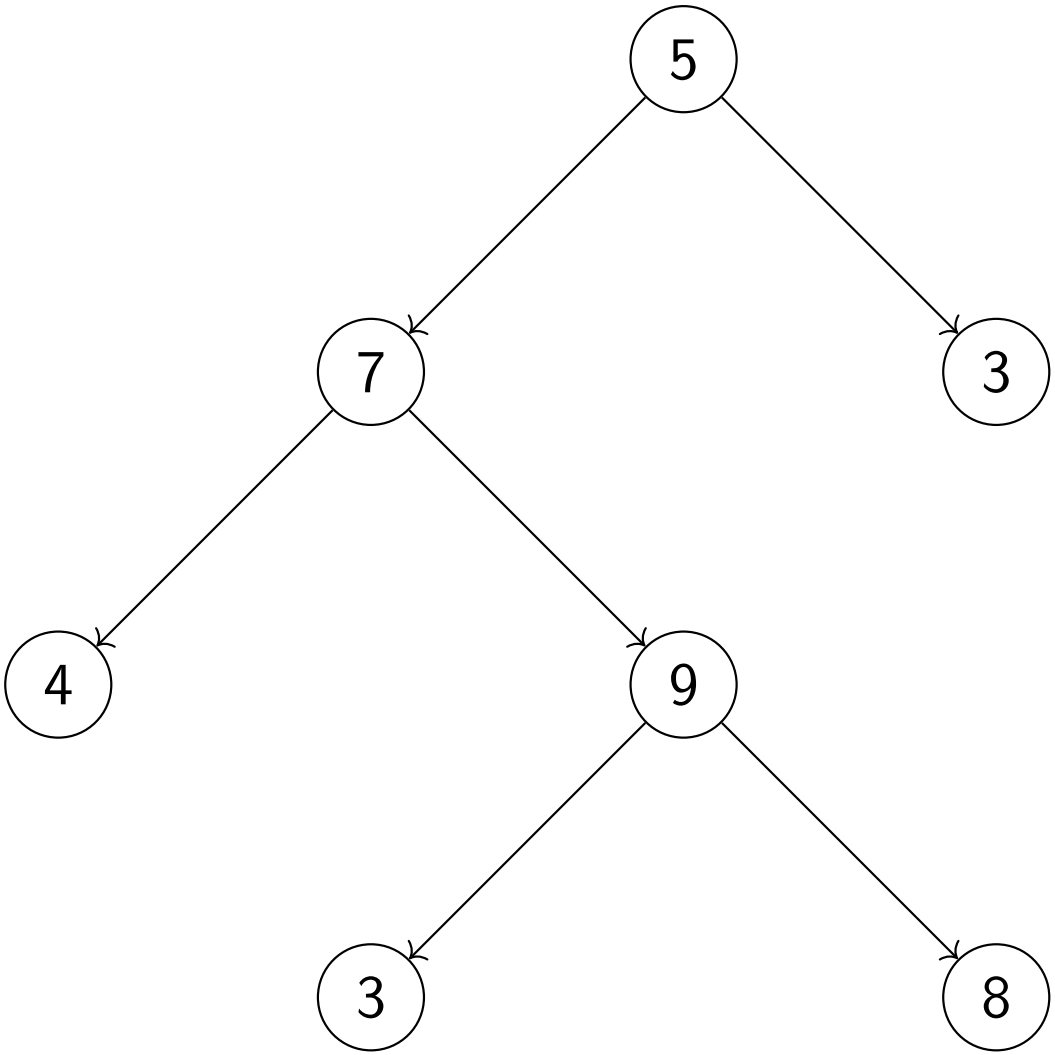
```
class Tree:
    def __init__(self, content, left=None,
        right=None):
         self.content = content
         self.left = left
         self.right = right
```

Trees

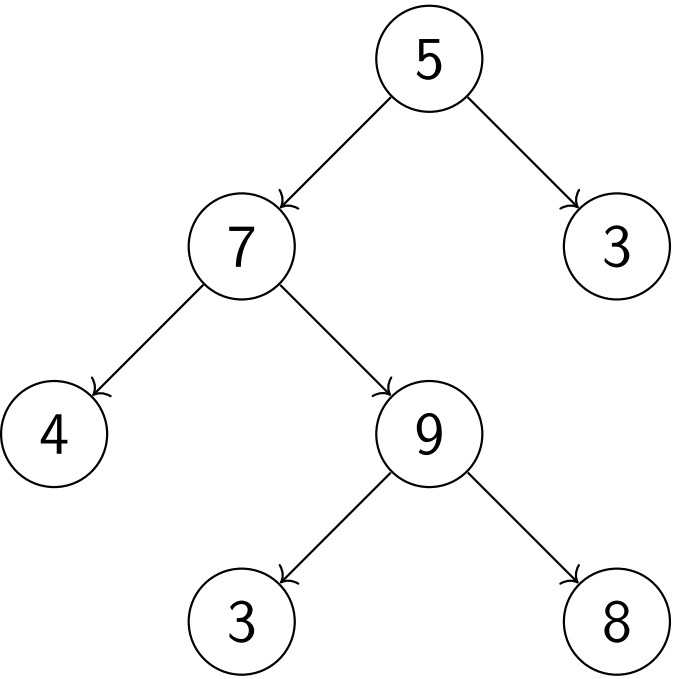
```
nodeD = Tree("D");
nodeD1 = Tree("D1");
nodeD2 = Tree("D2");

nodeD.left = nodeD1;
nodeD.right = nodeD2;
```



Trees



Trees




```
def funnyFunction(tree):
    if tree is None:
        return 0
    return - tree.content -
        funnyFunction(tree.left) +
        funnyFunction(tree.right)

print(funnyFunction(root))
```

Peer Feedback Exercise

- WAIT FOR ALL INSTRUCTIONS BEFORE YOU START MOVING!
- You will work in pairs
- Discuss your CW 8 with your partner
- Ask your partner what he/she is struggling with, and teach him/her
- Similarly, tell your partner what you are struggling with, and he/she will teach you
- If you are new to programming, find an experienced programmer as a pair
- If you are an experienced programmer, find someone that is new to programming as a pair

Outline

Stack again...

```
class Stack:
      def __init__(self):
          self.items = 10*[0];
          self.position = 0;

      def push(self, item):
           if (self.position < 10):
               self.items[self.position] = item;
               self.position = self.position + 1;
               return True;
           else:
               return False;

...
```

Stack again...

```
    def pop(self):
        if (self.position <= 0):
            return False;
        else:
            self.position = self.position - 1;
            return self.items[self.position];

stack = Stack();

stack.push(5);
stack.push(10);
stack.pop();
stack.pop();
print(stack.items[1]);
```

Stack again...

```
    def pop(self):
        if (self.position <= 0):
            return False;
        else:
            self.position = self.position - 1;
            return self.items[self.position];

stack = Stack();

stack.push(5);
stack.push(10);
stack.pop();
stack.pop();
print(stack.items[1]);
```

Why this code is wrong?

Private Members

```
class Stack:
      def __init__(self):
          self._items = 10*[0];
          self._position = 0;

      def push(self, item):
           if (self._position < 10):
               self._items[self._position] = item;
               self._position = self._position +
                   1;
               return True;
           else:
               return False;
...
```

Stack again...

```
def pop(self):
    if (self._position <= 0):
        return False;
    else:
        self._position = self._position - 1;
        return self._items[self.position];

def _checkStack(self):
    ....
```

Outline

Module

- Module: A collection of classes and/or functions
- Can be loaded with "import"

Module

Example

```
import random

generator = random.Random();

print (generator.random());
```

Library

- Library: a collection of modules
- Python Standard Library: https://docs.python.org/3/library/index.html

Standard Library

Exercise

1. Using the library, calculate the mean and standard deviation of a set of integers
2. Using the library, find a way to divide a string with "," into a list of integers. Example: "1, 2, 3, 4" should become [1, 2, 3, 4].
3. Using the library, find a way to randomly sample, without replacement, 3 items out of a set of integers

Library Location

- https://docs.python.org/3.6/library/sys.html
- Check sys.path

Installing Libraries

- pip install SomePackage
- pip3 install SomePackage
- python3 -m pip install SomePackage
- More information: https://docs.python.org/3/installing/index.html

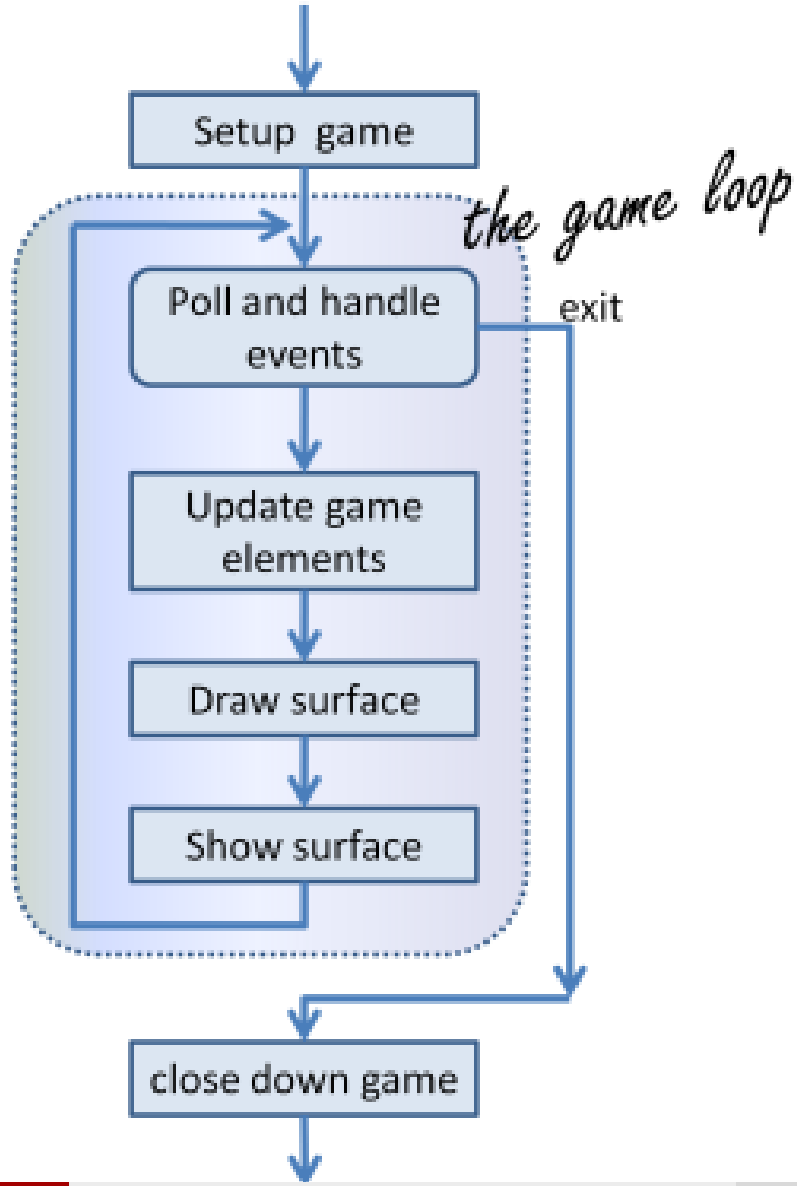
Outline

PyGame

PyGame

Why a game library for Data Science students?

Game Loop


Setup game
the game loop
Poll and handle events
exit
Update game elements
Draw surface
Show surface
close down game

Game Loop

```
import pygame

def main():
    pygame.init()
    surface_sz = 480

    main_surface =
       pygame.display.set_mode((surface_sz,
       surface_sz))

    small_rect = (300, 200, 150, 90)
    some_color = (255, 0, 0)

    while True:
        (...)
```

Game Loop

```
    while True:
        ev = pygame.event.poll()
        if ev.type == pygame.QUIT:
            break

        # Update your game objects and data
          structures here...

        main_surface.fill((0, 200, 255))
        main_surface.fill(some_color, small_rect)
        pygame.display.flip()

    pygame.quit()

main()
```

Display Graphics

- Load the file:

```
ball = pygame.image.load("ball.png")
```

Display Graphics

- Load the file:

```
ball = pygame.image.load("ball.png")
```

- Display the image:

```
main_surface.blit(ball, (100, 120))
```

Display Text

- Load the font:

```
my_font = pygame.font.SysFont("Courier", 16)
```

Display Text

- Load the font:

```
my_font = pygame.font.SysFont("Courier", 16)
```

- Create the text:

```
the_text = my_font.render("Hello, world!",
    True, (0,0,0))
```

Display Text

- Load the font:

```
my_font = pygame.font.SysFont("Courier", 16)
```

- Create the text:

```
the_text = my_font.render("Hello, world!",
    True, (0,0,0))
```

- Display the text:

```
main_surface.blit(the_text, (10, 10))
```

Everything Together

```
import pygame
import time
def main():
    pygame.init()     # Prepare the PyGame module
        for use
    main_surface = pygame.display.set_mode((480,
        240))
    # Load an image to draw.
    ball = pygame.image.load("ball.png")

    # Create a font for rendering text
    my_font = pygame.font.SysFont("Courier", 16)

    frame_count = 0
    frame_rate = 0
    t0 = time.clock()
```

Everything Together – 2

```
while True:
    ev = pygame.event.poll()
    if ev.type == pygame.QUIT:
        break

    frame_count += 1
    if frame_count % 500 == 0:
        t1 = time.clock()
        frame_rate = 500 / (t1-t0)
        t0 = t1

    main_surface.fill((0, 200, 255))
    main_surface.fill((255,0,0), (300, 100,
       150, 90))
    main_surface.blit(ball, (100, 120))
```

Everything Together – 3

```
            the_text = my_font.render("Frame = {0},
                rate = {1:.2f} fps"
                        .format(frame_count,
                            frame_rate), True, (0,0,0))
            main_surface.blit(the_text, (10, 10))
            pygame.display.flip()

        pygame.quit()

main()
```

Assignment

- Study Chapter 17 of How to Think Like a Computer Scientist (http://openbookproject.net/thinkcs/python/english3e/pygame.html)
- Study the Aliens game example (aliens.py)
- Propose and implement an "interesting" modification
  - For example: Exercise 4 at 17.10 (the variation where the shots could kill the player)
  - Any modification that is more trivial/easier than Exercise 4 will not get full marks, but will get some marks
- Write a "short" report (including your tests and your reflection)