

CSC131: Computational Thinking  
Homework Assignment # 4  
Due: Beginning of class on Wednesday, April 8, 2015.

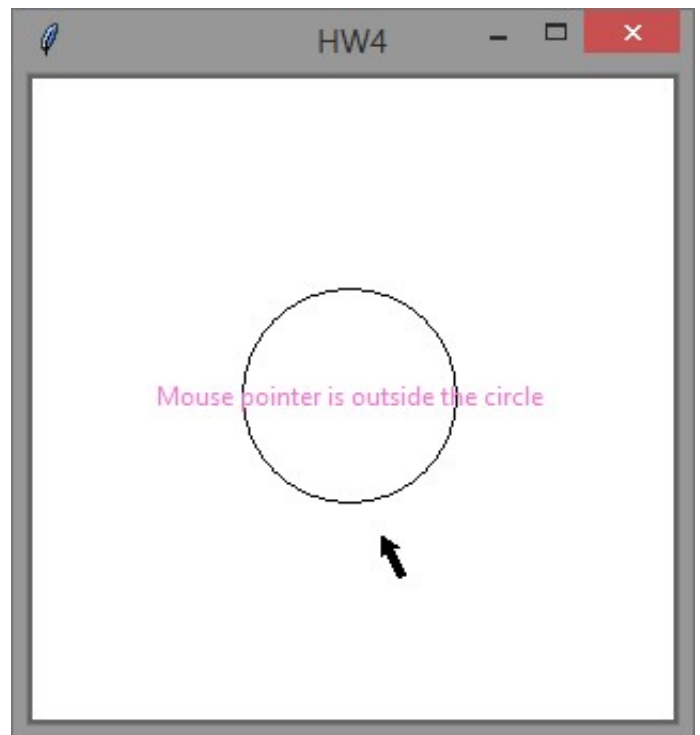
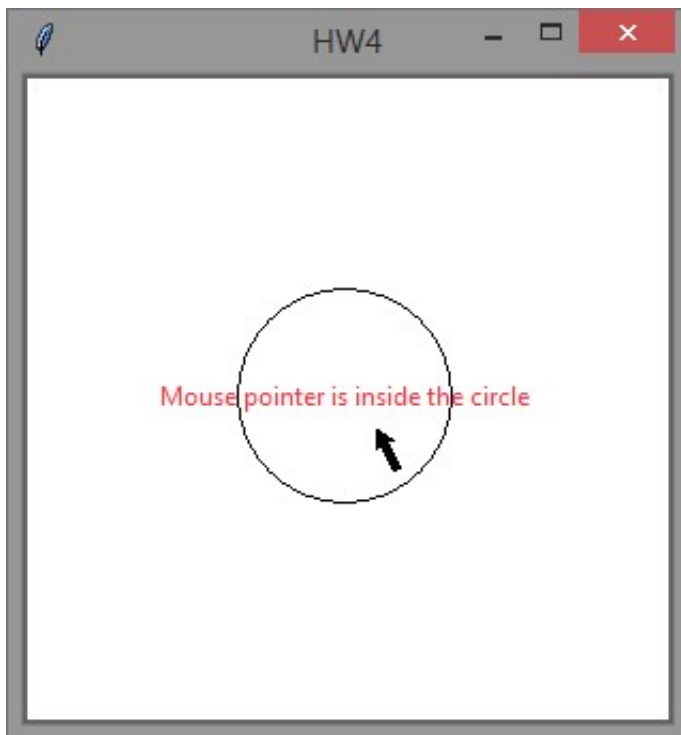
---

Write a Python program that draws a circle with radius 50 pixels and centered at the center of the canvas. Give the canvas white background color and make it expand with its frame whenever the user resizes the frame. The program will respond to some mouse and keyboard events as described below.

Whenever the left mouse button is pressed, display a message indicating whether the mouse pointer is inside the circle as shown in the figures below. The center of the message's baseline should be at the center of the circle. Moreover, the message should be displayed using a random color. You can use the following method to generate random colors:

```
def getRandomColor(self):
    """ Generates and returns a random color string such as '#c2f74a' """
    digits = "0123456789abcdef"
    color = "#"
    from random import randint
    for i in range(6):
        randomIndex = randint(0, 15)
        color += digits[randomIndex]

    return color
```

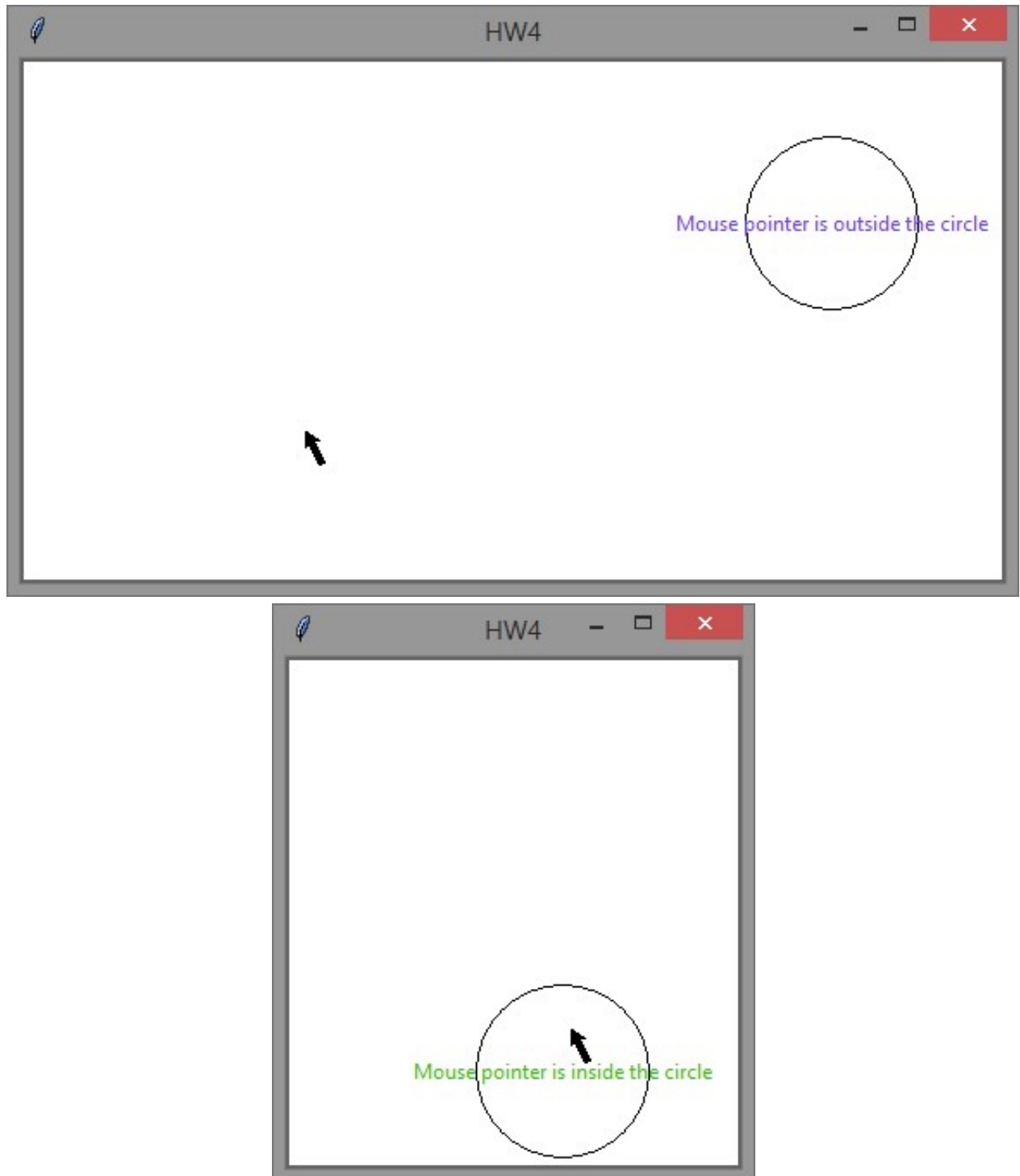


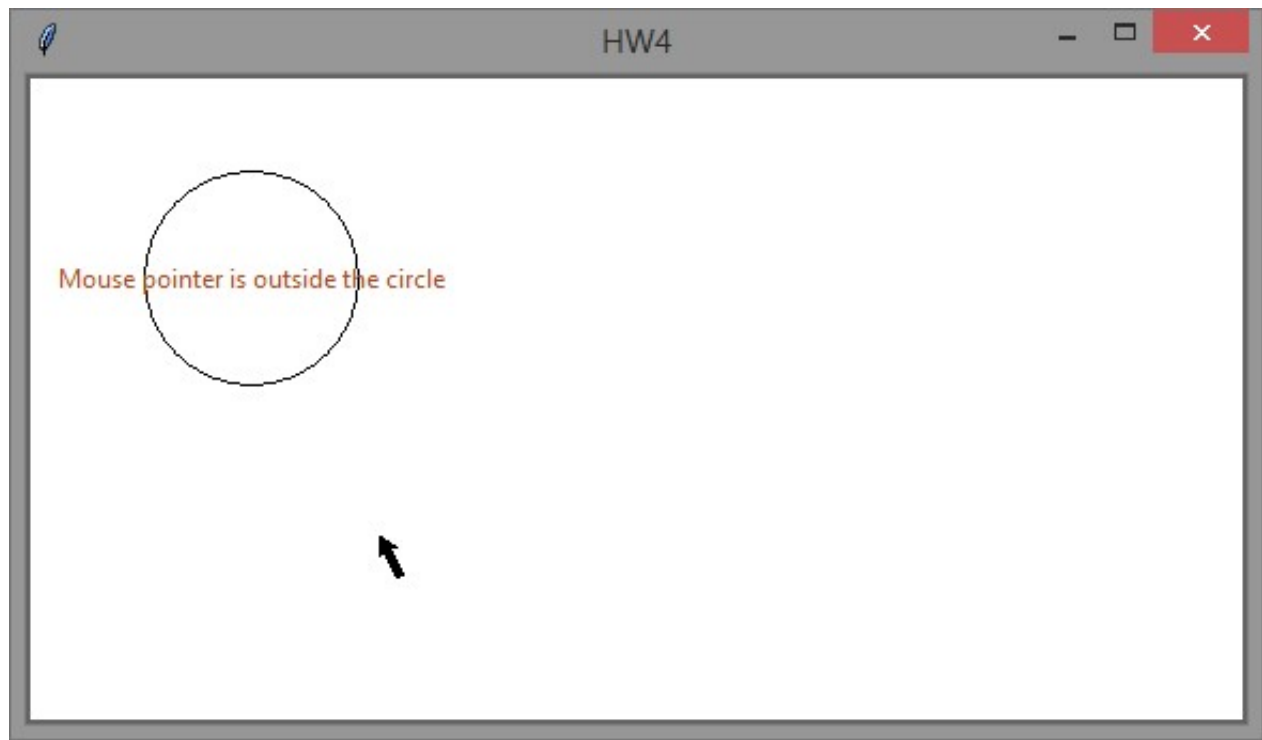
Your program also uses the arrow keys to move the circle left, right, up, or down when the Left arrow key, Right arrow key, Up arrow key, or Down arrow key is clicked. Every arrow-key click should move the circle 5 pixels in the appropriate direction. Delete any displayed text when the circle is moved. To bind the up arrow-key with the event handler `self.moveCircle()`, you can use the code:

```
self.canvas.bind("<Up>", self.moveCircle)

def moveCircle(self, event):
    """move circle up, down, left or right when user clicks an arrow key"""
    ...
    if event.keysym == "Up":
        self.canvas.move("circle", 0, -5)
    ...
    elif event.keysym == "Down":
    ...
```

Notice that using the arrow keys to move the circle, then clicking the left mouse button will always display the message at the center of circle. Examples are shown in this diagram





Please give your program the name `hw4.py`. Use comments to document and explain your code where needed. Make sure to upload an electronic copy of your source code in a folder named `HW\hw4` in your CSC 131 TRACE folder. Make sure to also turn in a stapled hard copy of the source code in class on the due date. Write your name and TRACE folder name at the top of your file.