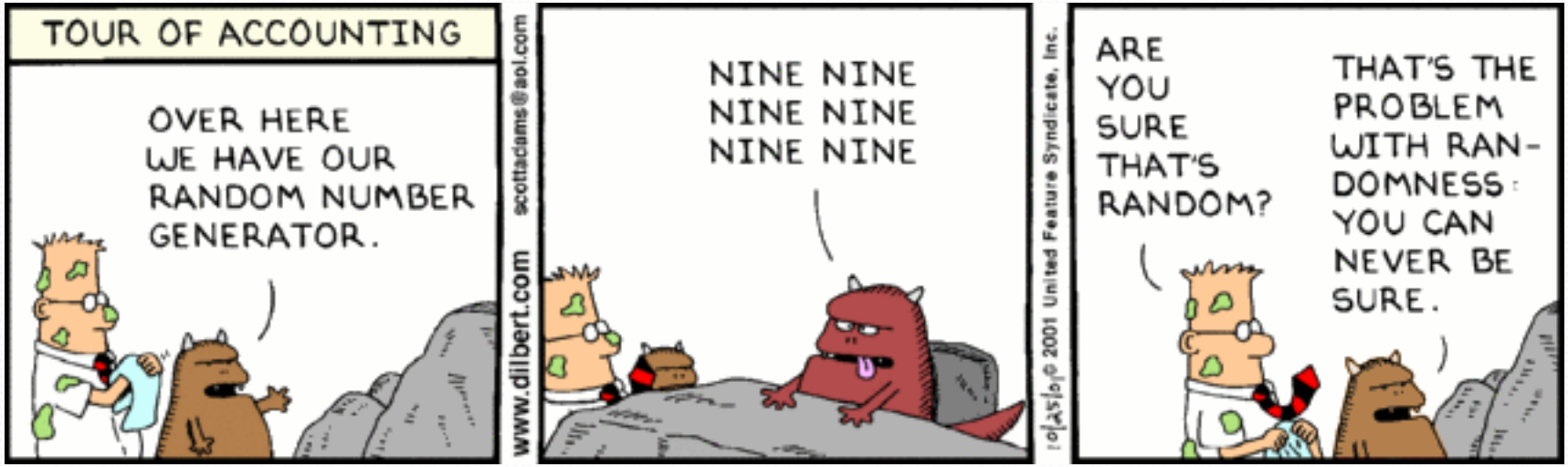
CS 115, Autumn 2021

Lecture 8: graphics; variables; random



Thanks to Marty Stepp and Stuart Reges for parts of these slides

Exercise:

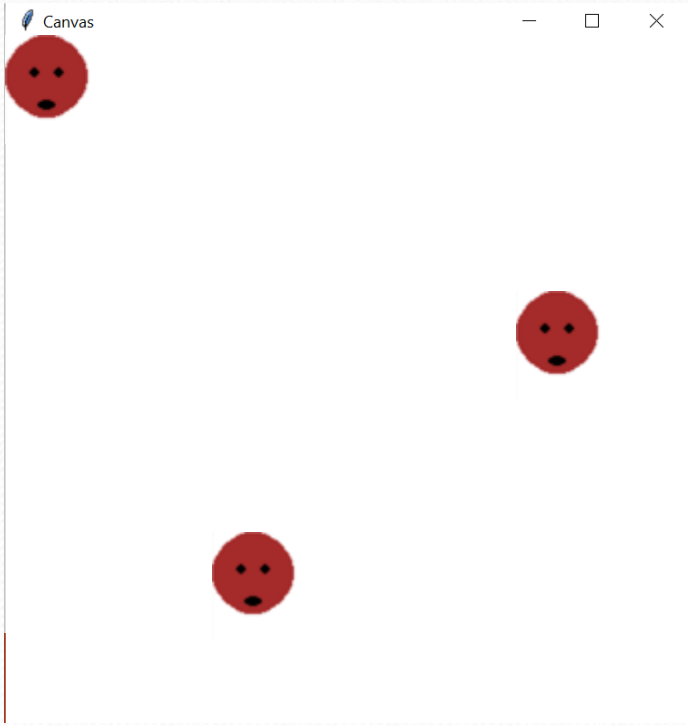
- Make the dot stay in the center of the window and increase in size as the mouse moves away from its center and decrease in size as it moves towards the center.

Drawing complex shapes

- What if we want a more complex shape to follow the mouse?

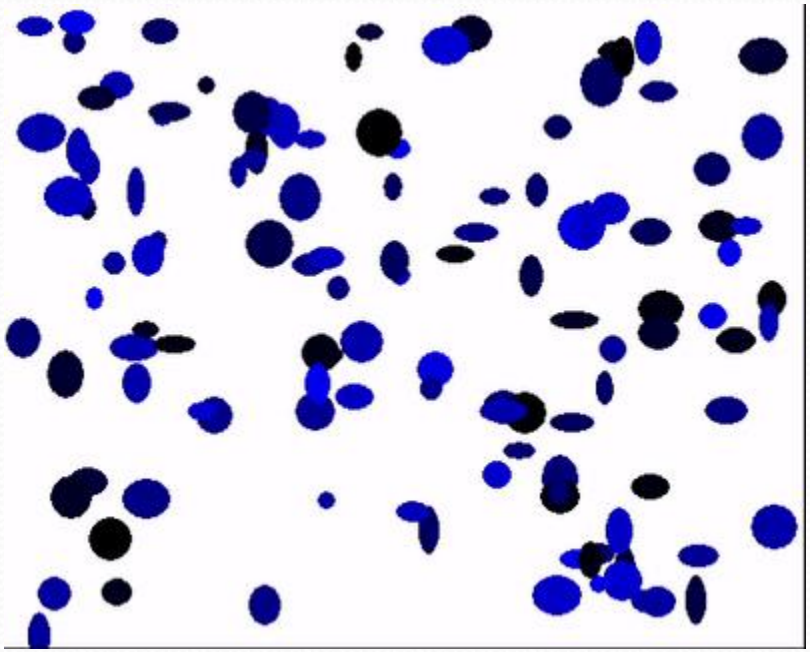
```
x = panel.get_mouse_x()
y = panel.get_mouse_y()
```

  - Can we just use `x` and `y` for all four shapes?
    - When we draw a shape, what part of the shape do the `x` and `y` get used for?



Making it rain

- What if we want to animate rain?
  - We could just draw a lot of blue circles in specific x, y locations
    - That wouldn't look very realistic – rain is much more random!



Pseudo-Randomness

- Computers generate numbers in a predictable way using a mathematical formula
- Function may include current time, mouse position
  - In practice, hard to predict or replicate
- True randomness uses natural processes
  - Atmospheric noise (http://www.random.org/)
  - Lava lamps (patent #5732138)
  - Radioactive decay

Random

- `random` generates pseudo-random numbers.
  - `random` can be accessed by including the following statement:

```
from random import *
```

Method name	Description
`random()`	returns a random float in the range [0, *1*) in other words, 0 inclusive to *1* exclusive
`randint(`***min, max***`)`	returns a random integer in the range [min, *max*) in other words, min to *max*-1 inclusive

  - Example:

```
from random import *
random_number = randint(1, 10)    # 1-9
```

Generating random numbers

- To get a number in arbitrary range [*min*, *max*] inclusive:

  `randint(`***min, max***`)`

  - Where ***size of range*** is (***max*** `-` ***min*** `+ 1`)

  - Example: A random integer between 4 and 10 inclusive:

    ```
    n = randint(4, 10)
    ```

Exercise: rain

- Write a program that draws a new oval on a `drawing_panel` every 10ms. All ovals should be:
  - At random x locations
  - At random y locations
  - Between 10 and 30 pixels wide
  - Between 10 and 30 pixels tall
  - Be random colors with 0 red, 0 green and a random amount of blue