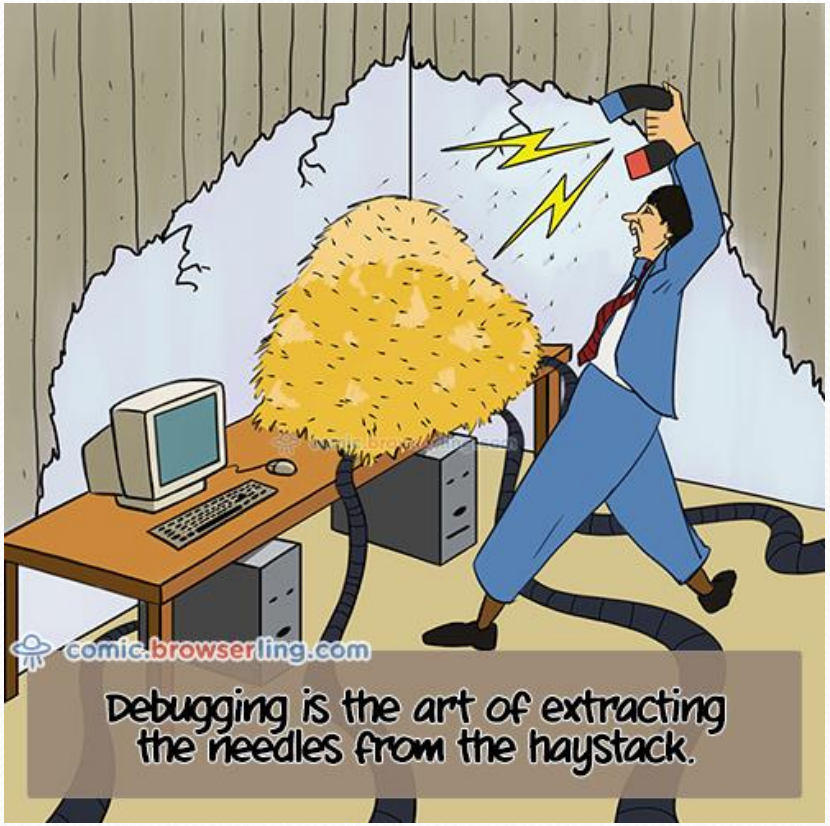
CS 115, Autumn 2021

Lecture 23: parameters



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

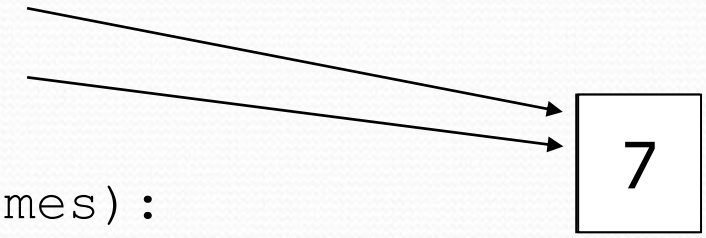
How parameters are passed

- When the function is called:
  - The value is stored into the parameter variable.
  - The function's code executes using that value.

```
def main()
    chant(3)
    chant(7)

def chant(times):
    print("Just", times, "salads...")

main()
```



Common errors

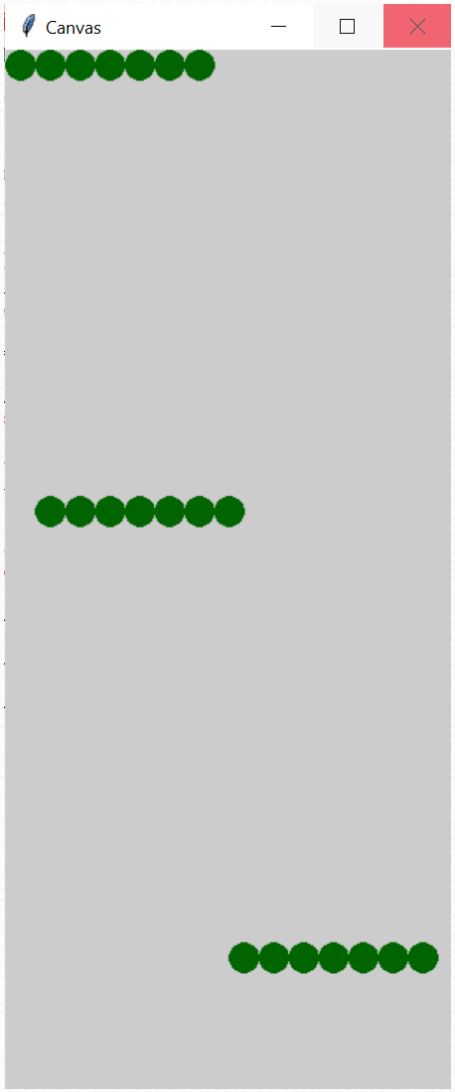
- If a function accepts a parameter, it is illegal to call it without passing any value for that parameter.

```
chant()          // ERROR: parameter value required
```

- Exercise: Change the lab 5 code to place the bird drawing in its own function.

Exercise: `green_circles`

- Alter the code used to produce the picture to the left which contains a separate function for each line of green circles to just contain one green circle drawing function.



Green circles initial code

```
def main():
    panel = drawing_panel(300, 700, "#CCCCCC")
    green_circles_top(panel)
    green_circles_middle(panel)
    green_circles_bottom(panel)

def green_circles_top(panel):
    for i in range(0, 7):
        panel.fill_oval(i * 20, 0, 20, 20, "green")

def green_circles_middle(panel):
    for i in range(0, 7):
        panel.fill_oval(20 + i * 20, 300, 20, 20, "green")

def green_circles_bottom(panel):
    for i in range(0, 7):
        panel.fill_oval(150 + i * 20, 600, 20, 20, "green")

main()
```

Green circles initial code

```
def main():
    panel = drawing_panel(300, 700, "#CCCCCC")
    green_circles(panel, 0, 0)
    green_circles(panel, 20, 300)
    green_circles(panel, 150, 600)

def green_circles(panel, x, y):
    for i in range(0 7):
        panel.fill_oval(x + i * 20, y, 20, 20, "green")

main()
```

Parameters and loops

- A parameter can guide the number of repetitions of a loop.

```
def main():
    chant(3)

def chant(times):
    for i in range(1, times):
        print("Just a salad...")
```

Output:

```
Just a salad...
Just a salad...
Just a salad...
```

Value semantics

- **value semantics**: When value types (`int`, `double`) are passed as parameters, their values are copied.
  - Modifying the parameter will not affect the variable passed in.

```
def strange(x)
    x = x + 1
    print("1. x =", x)

def main()
    x = 23
    strange(x)
    print("2. x =", x)
    ...

main()
```

Output:

```
1. x = 24
2. x = 23
```

A "Parameter Mystery" problem

```
def main()
    x = 9
    y = 2
    z = 5

    mystery(z, y, x)

    mystery(y, x, z)
```

```
def mystery(      x,          z,          y):
    print(str(z), & "and", str(y - x))


main()
```

Other types as parameters

```
def main():
    say_hello("Merlin")
    cat = "Sir Purrcival"
    say_hello(cat)

def say_hello(name):
    print("Welcome,", name)
```

Output:

```
Welcome, Merlin
Welcome, Sir Purrcival
```

- Modify the `green_circles` program to use color parameters.

Green circles solution

```
def main():
    panel = drawing_panel(300, 700, "#CCCCCC")
    green_circles(0, 0, 7, "green")
    green_circles(20, 300, 4, "yellow")
    green_circles(150, 600, 12, "#CC5555")

def green_circles(panel, x, y, count, color):
    for i in range(0, count + 1):
        panel.fill_oval(x + i * 20, y, 20, 20, color)

main()
```