

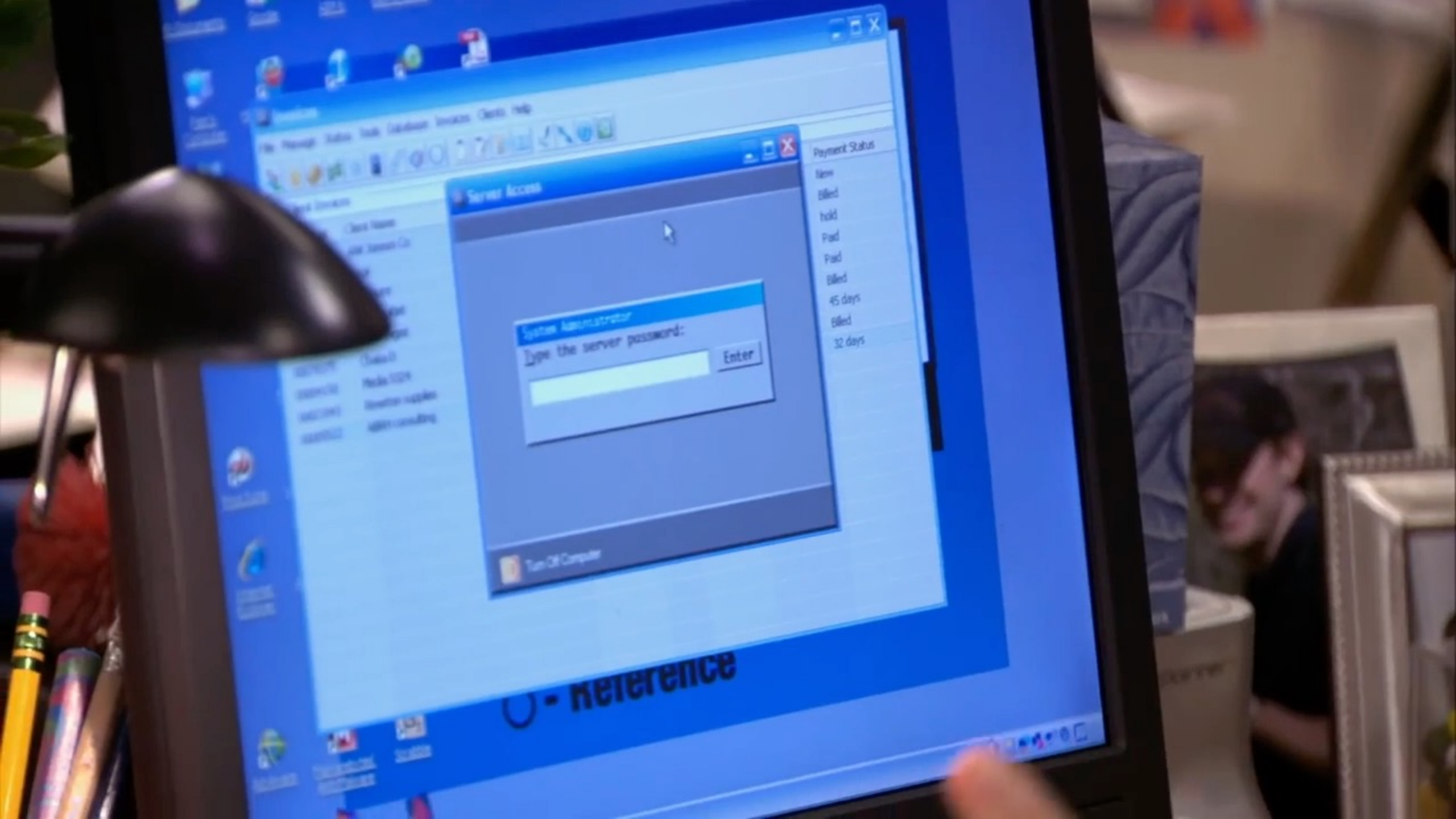
ATTACK ORIENTED PROGRAMMING

IoT Security and Privacy Lab, Florida Tech

Objectives

Intro to attack oriented programming

- Introduce the python programming language, the concepts of variables, selection, iteration, and execution.
- Discuss a brute force attack as a method of iterating through all possible inputs to identify a solution.
- Write your first python code to conduct a brute force attack.



Server Access

System Administrator

Type the server password:

Enter

Reference

Brute Force

```
for guess in range(0,999):  
    print(guess)
```

When a problem contains a set of limited inputs, we can use a computer's ability to brute force the solution by trying all possible inputs.



Intro to Programming

```
print("Spongebob SquarePants")
```



`print()` is a function call. It outputs the contents passed in the **parameter** to your computer screen



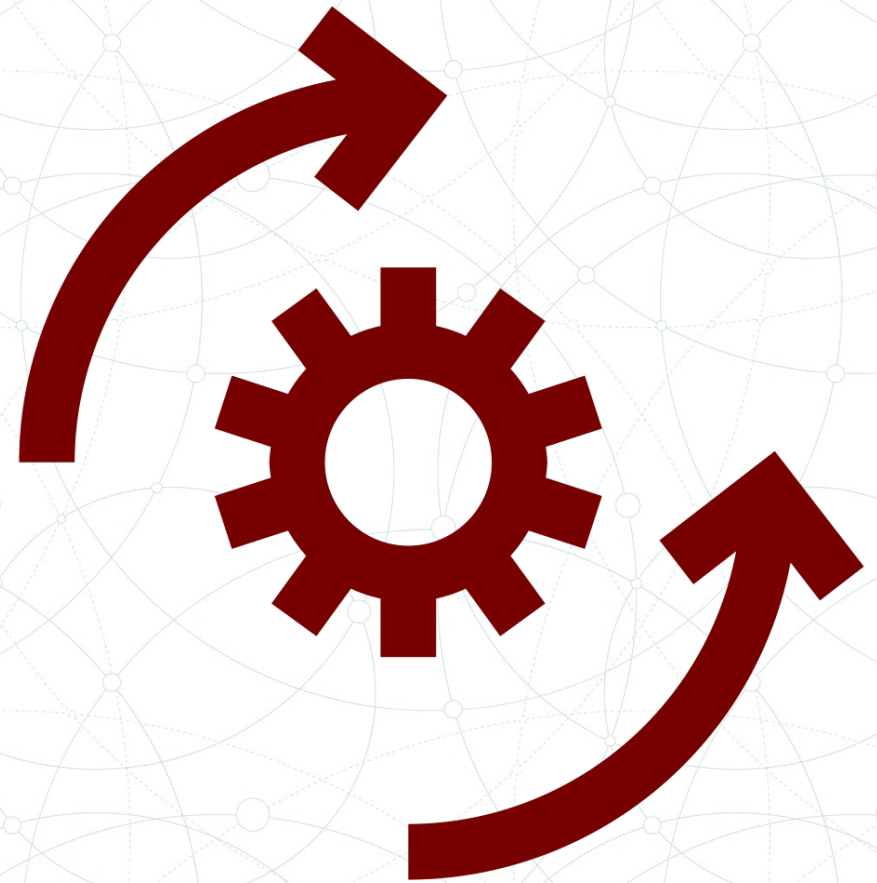
Variables

```
name = input("What is your name >>> ")  
name_length = len(name)  
print("Length of your name:", name_length)
```



`name` and `name_length` are both variables. `name` stores the letters (characters) that make up your name. `name_length` stores a number (integer) that holds the length of your name.

Check on learning: what is `len()`?



Iteration

```
import time, sys

name = input("What is your name >>> ")
name_length = len(name)

for count in range(0, name_length):
    print(name[count], end='')
    sys.stdout.flush()
    time.sleep(0.25)
```

for count in range(0, name_length) loops (iterates) through the characters in your name and then executes the statements in the loop body, which slowly prints each letter to the screen.



Selection

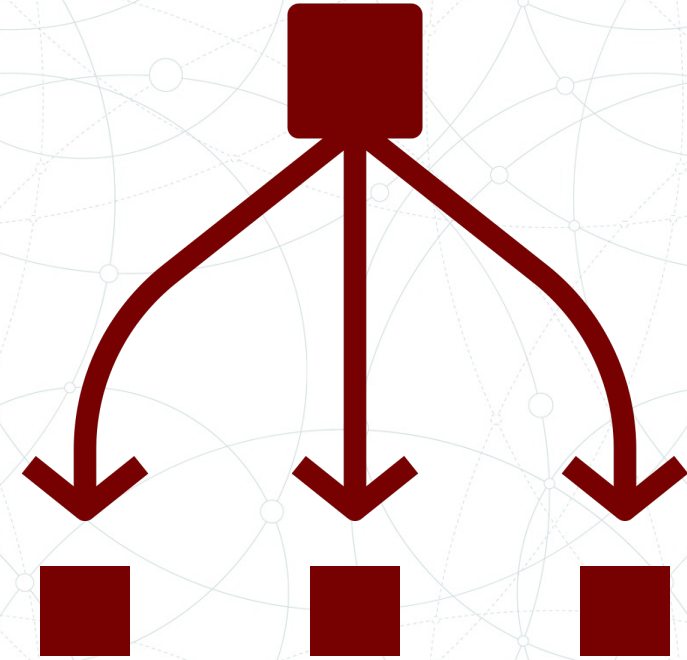
```
import time, sys

name = input("What is your name >>> ")
name_length = len(name)
new_name = ''

for count in range(0, name_length):
    if (count % 2 == 0):
        new_name += name[count].upper()
    else:
        new_name += name[count].lower()

print(new_name)
```

`if (count % 2 == 0)` checks (selection) to see if the count is even, if even, it appends an uppercase letter, otherwise it just appends a lowercase letter.



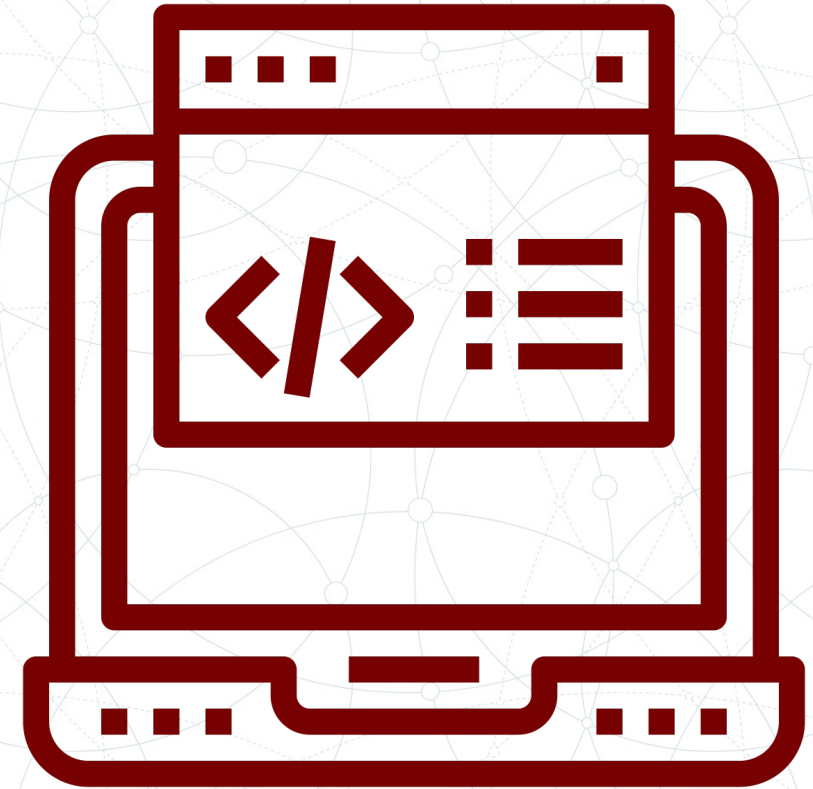
Execution

```
import time, sys, os

name = input("What is your name >>> ")
name_length = len(name)
new_name = ''

for count in range(0, name_length):
    if (count % 2 == 0):
        new_name += name[count].upper()
    else:
        new_name += name[count]

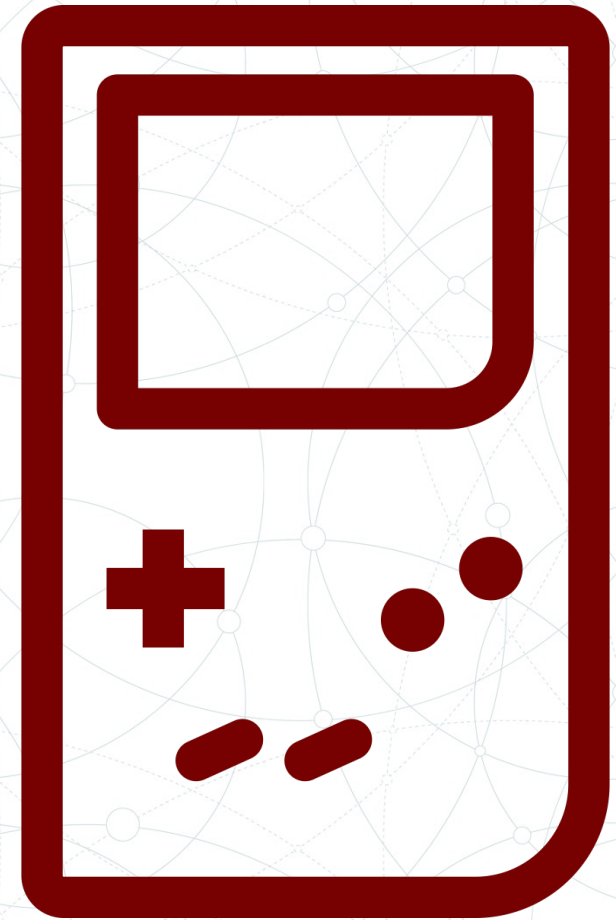
cowsay = os.popen("cowsay " + new_name)
cowsay_output = cowsay.read()
print(cowsay_output)
```



`os.popen("cowsay " + new_name)` execute the program cowsay with your name as the argument. `cowsay.read()` reads the output of the program `print(cowsay_output)` now prints the output of the cowsay program.

Mom's Ransomware

- Play the Gameboy to learn why mom locked your computer
- Write a program to brute-force the pin to get the password
- Go to <https://10.3.141.1/> to connect
- Play through the game on the GameBoy to get the full story



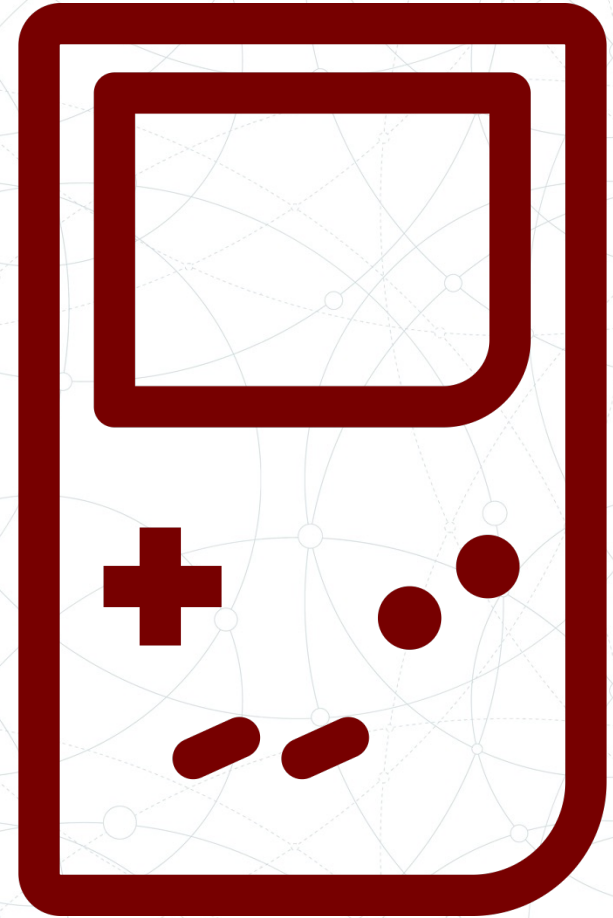
Programming Review

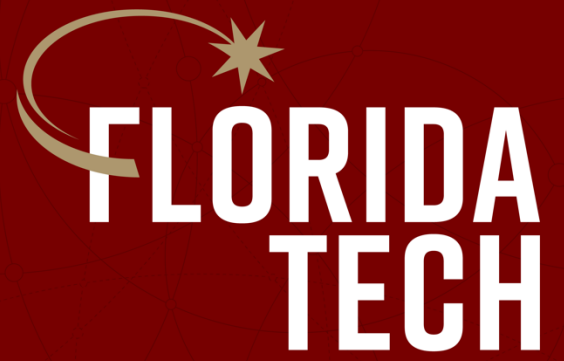
```
import os

name = input("What is your name >>> ")
name_length = len(name)
new_name = ''

for count in range(0, name_length):
    if (count % 2 == 0):
        new_name += name[count].upper()
    else:
        new_name += name[count]

cowsay = os.popen("cowsay " + new_name)
cowsay_output = cowsay.read()
print(cowsay_output)
```





Thank you.