

Scoping + Indentation

~~Scope~~ - where a variable is valid to use.

Indentation

Ex 1

```
if age >= 16:
    inside ← { print("Drive")
    → print("Go to park")
```

age 20
Drive
Go to Park

age 10
Go to Park

Ex 2

```
if age >= 16:
    print("Drive")
    print("Go to park")
```

→ inside if
age 20 age 10
Drive
Go to Park

Scope - a variable is "alive" or "valid" from the 1st place it is used until that level of indentation runs out.

→ If we move this here we can use count "longer".

if

count = 0

{ count 0/2

if

| count 0/2

else:

| count 0/2

else:

→ Can't use count!

stmtX } Can't use count.

count

→ count's scope is here

$$2S = n(n+1)$$

$$S = \frac{n(n+1)}{2}$$

for loop version 1:

for var in range(n):

sets var = 0, 1, 2, ... n-1

↓
Runs code

an + int

for loop version 2:

for var in range(a, b):

sets var = a, a+1, a+2, ... b-1

↑
exclusive

any int
any int

total 0 x 3/6

total = total + (i+1)

i 0, 1, 2, 3, ... , n-1

↓ ↓ ↓
1 2 3

↑ i+1

S = 1 + 2 + 3 + 4 + ... + n

S = n + (n-1) + (n-2) + ... + 1

2S = (n+1) + (n+1) + (n+1) + ... + (n+1)

for version 3

for var in range(start, end, skip):

var = start, start + skip, start + 2 * skip,
start + 3 * skip, ... < end

var = var + something

SHORTHAND

var += something

↑

any operator