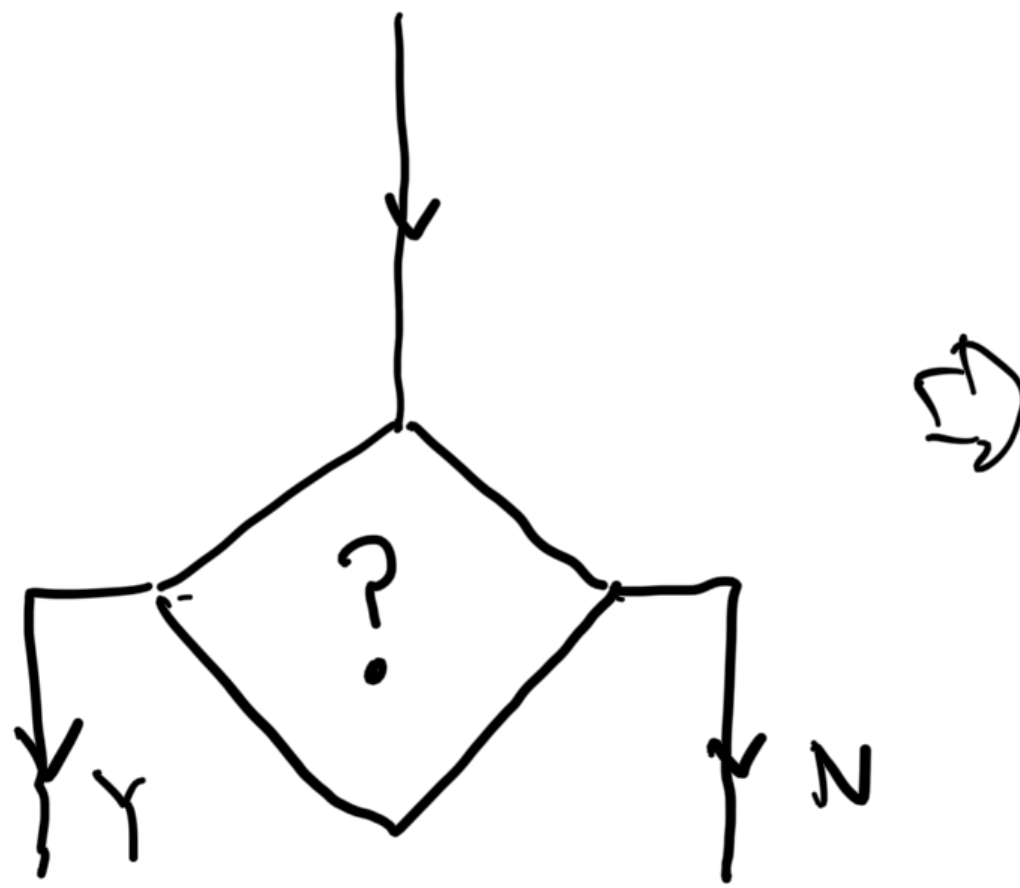


Branching & Flow Control



if condition:



else:



Sometimes, the "N" branch is just
"do nothing" ...

if condition:



The point: The if-else construct is
the general case, and the if construct

is just a special "else = do nothing" case.

What about "condition"?

→ condition equates to a **BOOLEAN!**
i.e. True or False.

Ex:

a == 4

→ equates to T if a = 4

→ equates to F if not.

"==" is not the same as "="

logical
Comparison

assignment

a = 4

⇒ store "4" in
the memory location
corresponding to a.

Loops

① For loops.

↗ I know how many times I want to execute the loop.

for i in range(10):

10 times { i=0
i=1
i=2
⋮
i=9 } ↓

l = ["a", "b", "c"]

for item in l:

3 times ↓ item="a"
item="b"
item="c"

② While loops.

↗ I don't know that!!

condition!
while a < 4:

⋮
change a!
⋮

loop ends when condition is no longer met!

Occasionally, we have multiple conditions

in a single condition loop.

for executing, making a loop

Example: Loop through the elements of a list, until the end, unless we reach a list element that equals "end", at which point we exit the loop.

l = ["bob", "alice", "fred", "end", "jane"]

for item in l:

if item == "end":

break

print(item)
