

# 4 Branching (Control)

Monday, February 12, 2024 2:38 PM

• Boolean True False

• Boolean Expressions.

== equality

!=

relational operators

< <=  
> >=

logic connectives


or and not.


• Control

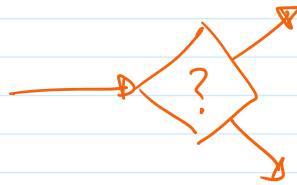
A program can make decisions according to the value of variables.

- control execution of different pieces of code.
- execute some code conditionally

If some conditions are true

do blah 

else do blah blah 



• (1) IF

Syntax

≡

if cond:  
  blocks

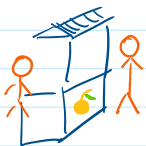
≡  
≡  
≡  
≡

Note: block is indented

A "Block" of code

- a sequence of statements
- begins when indentation increases
- ends when indentation decreases
- Blocks can contain other Blocks.

• Demo:



- Lemonades are 1.25 each
- if somebody buys more than 6 there is a 10% discount.

• (2) IF

```
≡≡≡
if cond :
  block1
else :
  block2
≡≡≡
```

- if cond evaluates to true, block1 is executed otherwise block2 is executed.

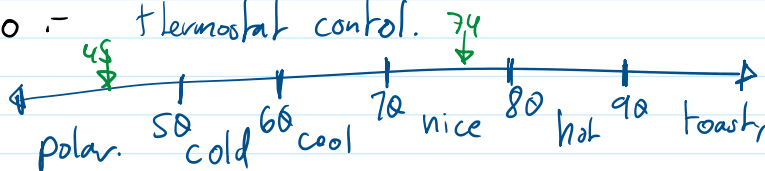
• Demo: Check if a number is even or odd:

• (3) IF

```
≡≡≡
if cond1 :
  block1
elif cond2 :
  block2
elif cond3 :
  block3
...
else :
  block_def
≡≡≡
```

- conditions are tested sequentially, until one of them is true. then the corresponding block is executed. then program resumes after the if statement.

• Demo :- thermostat control.



• MORE BOOLEAN OPERATORS

x in seq     .- True if x is in seq  
x not in seq   .- True if x is not in seq

Common operation, check whether an element is inside a sequence type.

Note. can be expensive if the sequence is large.

More:

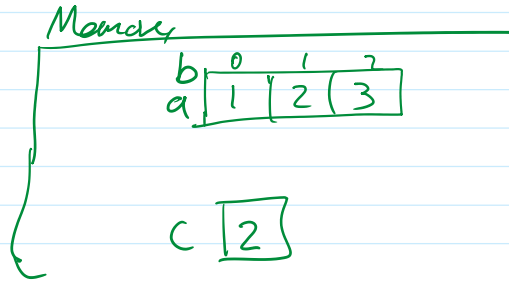
is  
not is.

$x$  is  $y$  :- true if both  $x$  &  $y$  refer to the same object in memory.  
or they are the same non sequence value.

$a = [1, 2, 3]$

$b = a$

$c = 2$



## • THE CONDITIONAL EXPRESSION

Motivation:

if  $x < y$ :

$x = x + 1$

else

$x = x - 1$

block modifies same variable.

a shorthand

$x = x + 1$  if  $x < y$  else  $x - 1$

Syntax:

$expr_1$  if  $cond$  else  $expr_2$

takes the value  $expr_1$  if  $cond$  is true  
takes the value  $expr_2$  otherwise.

## • CONDITIONALS CAN BE NESTED

```
if n%3 == 0:
    if n%5 == 0:
        print('FizzBuzz')
    else:
        print('Fizz')
else:
    if n%5 == 0:
        print('Buzz')
    else:
        print('-o-')
```