

4 Branching (Control)

Friday, September 20, 2024 8:17 AM

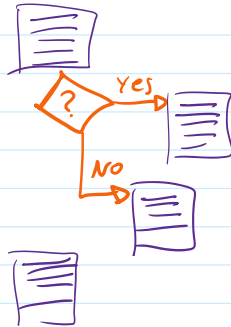
• BOOLEAN

True False.
Boolean operators
== < <= or and
!= > >= not

• CONTROL

A program can make decisions according to the values of its variables.

- Control execution of different parts of code



• (1) IF

Syntax:

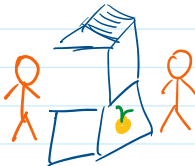
```
IF cond :  
    block
```

A block of code.

- A sequence of code statements
- begin when indentation increases
- ends when indentation decreases
- blocks can contain other blocks.

• Demo:

Problem:



- lemonades are \$ 1.25 each
- if the client buys more than six lemonades the client receives a 8% discount.
- write a program that computes the total.

Drafts:

- 1.- Set price of lemonades
- 2.- Ask client for quantity

- 1.- Set price of lemonades
- 2.- Ask client for quantity
- 3.- Compute total
- 4.- if quantity is greater than 6
 apply discount
- 5.- display total.

• (2) IF

```

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

```

if cond is 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 default.
≡≡≡

```

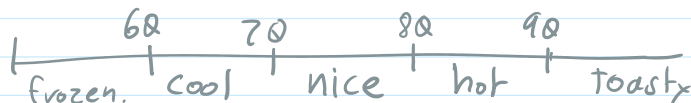
- conditions are tested sequentially until one of them is true. then only the corresponding block is executed.

- the program resumes after the if statement

Demo:-

Thermometer.

Given a numeric input you want to describe the temperature.



• MORE BOOLEAN OPERATORS

x in seq

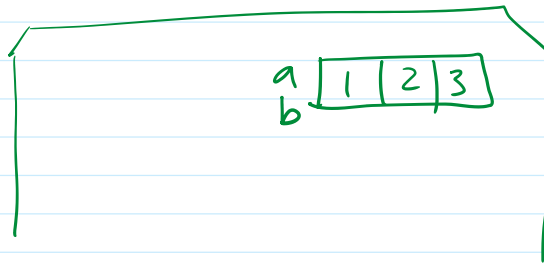
True if x is an item in seq

x not in seq true if x is not an item in seq
very common operation

Note: this operation is expensive if the sequence is large

x is y true if both x, y refer to the same entity in memory.
 x is not y

e.g. $a = [1, 2, 3]$
 $b = a$
 a is b true.



THE CONDITIONAL EXPRESSION

Motivation:
if $x < y$:
 $x = x + 1$
else:
 $x = x - 1$
both blocks modify same variable.

Shorthand:

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

Syntax

$expr_1$ if $cond$ else $expr_2$

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

CONTROL CAN BE NESTED

Demo: "FizzBuzz"

Given a positive integer n , print "Fizz" if the number is divisible by 3, "Buzz" if the number is divisible by 5, "FizzBuzz" if the number is divisible by both 3 & 5, otherwise print the number.

```
n = int(input('number? '))
```

```
if n % 3 == 0:  
    if n % 5 == 0:  
        print('FizzBuzz')
```

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

—•— Eof.