

Lecture6 design

abstraction

functional abstractions

name some computational process and see them as a whole, without caring its inner mechanics.



Functional Abstractions

```
def square(x):  
    return mul(x, x)
```

```
def sum_squares(x, y):  
    return square(x) + square(y)
```

What does `sum_squares` need to know about `square`?

- Square takes one argument. Yes
- Square has the **intrinsic** name `square`. No
- Square computes the square of a number. Yes
- Square computes the square by calling `mul`. No

```
def square(x):  
    return pow(x, 2)
```

```
def square(x):  
    return mul(x, x-1) + x
```

If the name “square” were bound to a built-in function,
`sum_squares` would still work identically.

- guidelines for the names of the functions.

Choosing Names



Names typically *don't* matter for correctness

but

they matter a lot for composition

From:

true_false

d

play_helper

To:

rolled_a_one

dice

take_turn

Names should convey the *meaning or purpose* of the values to which they are bound.

The type of value bound to the name is best documented in a function's docstring.

Function names typically convey their effect (`print`), their behavior (`triple`), or the value returned (`abs`).

5

which values deserves a name?

- repeated compound expressions
- meaningful parts of complex expressions
 - for example: discriminant in equation solving

more tips

- names can be long if they help document your code
 - `average_age` instead of `aa` or `sth else`
- names can be short if they represent generic quantities
 - `x, y, z--` real numbers
 -

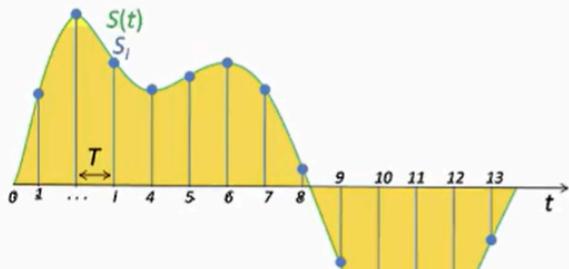
WHY do we need higher order functions?

function example: sound

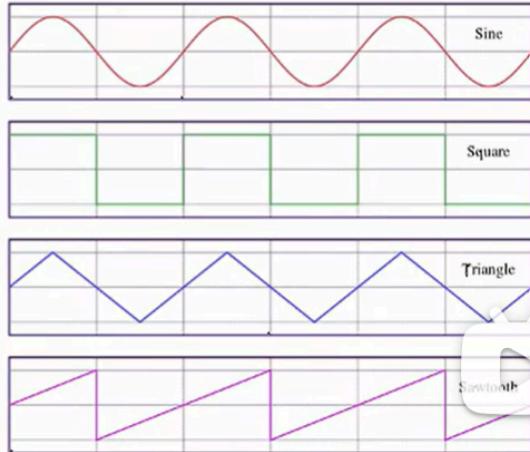
WAV Files

WAV Files

The Waveform Audio File Format encodes a sampled sound wave



A triangle wave is the simple waveform with the most pleasing sound



so that's why we're gonna use it in this demo

https://en.wikipedia.org/wiki/Triangle_wave

[https://en.wikipedia.org/wiki/Sampling_\(signal_processing\)](https://en.wikipedia.org/wiki/Sampling_(signal_processing))

[!info]Q&A session

about inequality in python of float

$1/3 + 1/3 + 1/3$