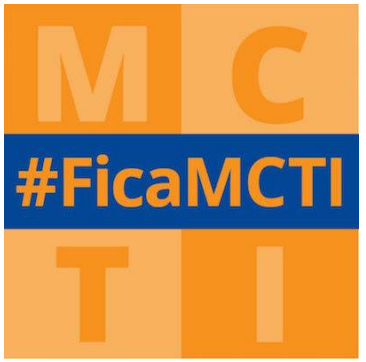
python

Cosmology with python:

Beginner to Advanced in one week

Tiago Batalha de Castro



What is Python? (From python.org)

- Python is an interpreted, object-oriented, high-level programming language with dynamic semantics
- It is very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together
- Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance
- Python supports modules and packages
- It is Free

Python is an interpreted, object-oriented (...)

- Python does not require compilers.
  - Easier to debug
    - No waste of time in the eternal loop code → compile → check → code ...
  - But also easier to write bad codes
    - No warnings, clues, or tips
- **About Object-oriented programming: well… We will see what it means in this class**

Procedure-Oriented Programming Paradigm (1)

- Complete a task following structured steps based on **procedures** (statements and functions):
  - For a competent procedure-oriented programmer it shouldn’t be hard to identify sub-tasks that should be written as a function in order to write good modulated codes.
  - For instance, on a code for SNe analysis the routine to calculate the distance modulus best fit in a function, while parameters that define my model best fit a simplistic variable.

Procedure-Oriented Programming Paradigm (2)

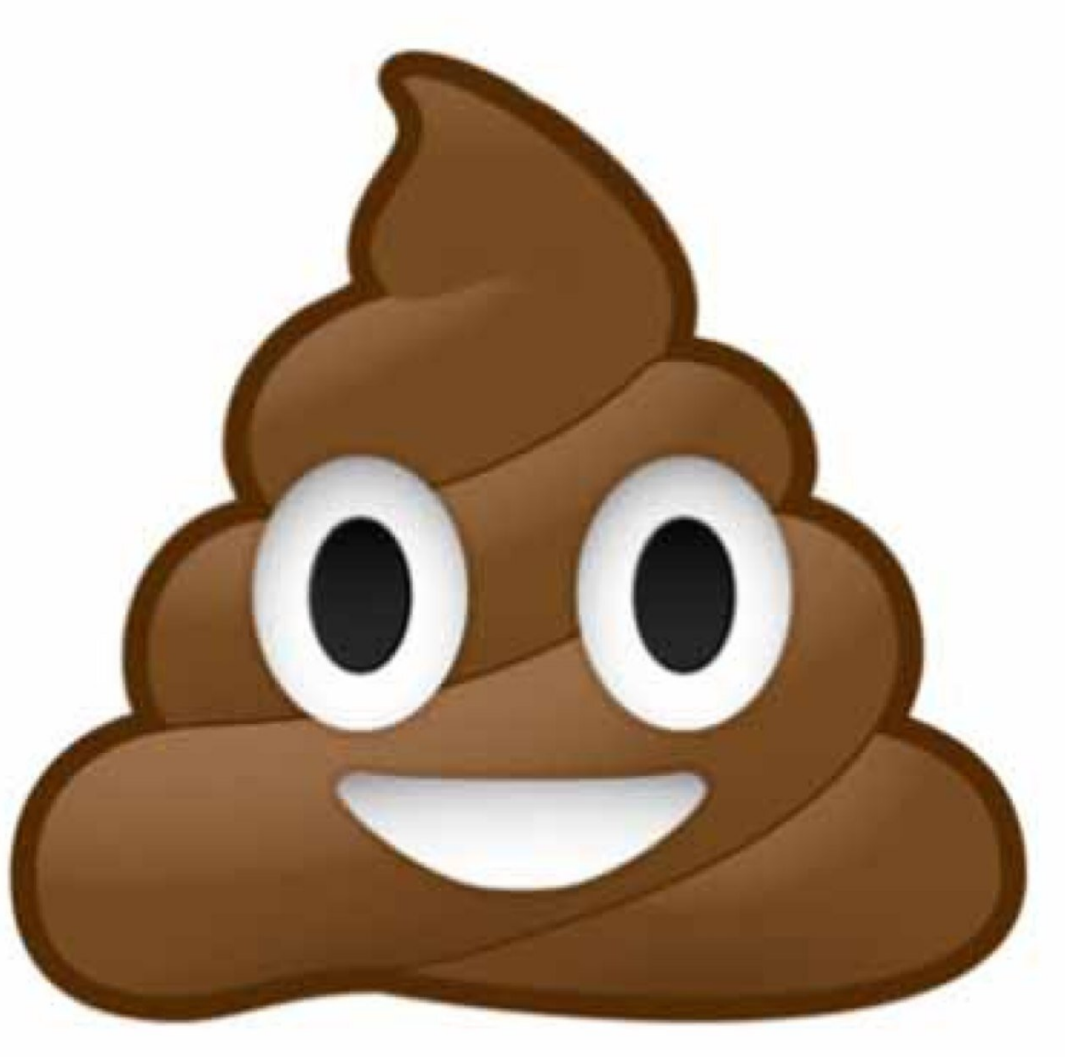
- On a POPP what is a Cosmological Model?
  - It is obviously more complex than a function:
    - A cosmological model has many **methods** to compute different things, like: distances, background evolution, perturbation evolution…
  - It has different **attributes** that define them:
    - For instance, LCDM predictions are based on the values of the density of the different components of Universe, the Hubble constant…
  - Once you define the **attributes** to a cosmological model you create an **instance** of this cosmological model:
    - Different values of $\{\Omega_m, \Omega_\Lambda, H_0\}$ define different instances of LCDM

Procedure-Oriented Programming Paradigm (3)

- On a POPP, what is a Cosmological Model?
  - A Cosmological Model does not fit on a POPP!
- For a Cosmological Model fit inside a programming paradigm you would need a built-in structure that has intrinsic **Methods and Attributes.**
- A whole bunch of other **classes** require the same structure

Python Object Oriented Programing

Python Object Oriented Programing



Object-Oriented Programing Paradigm (1)

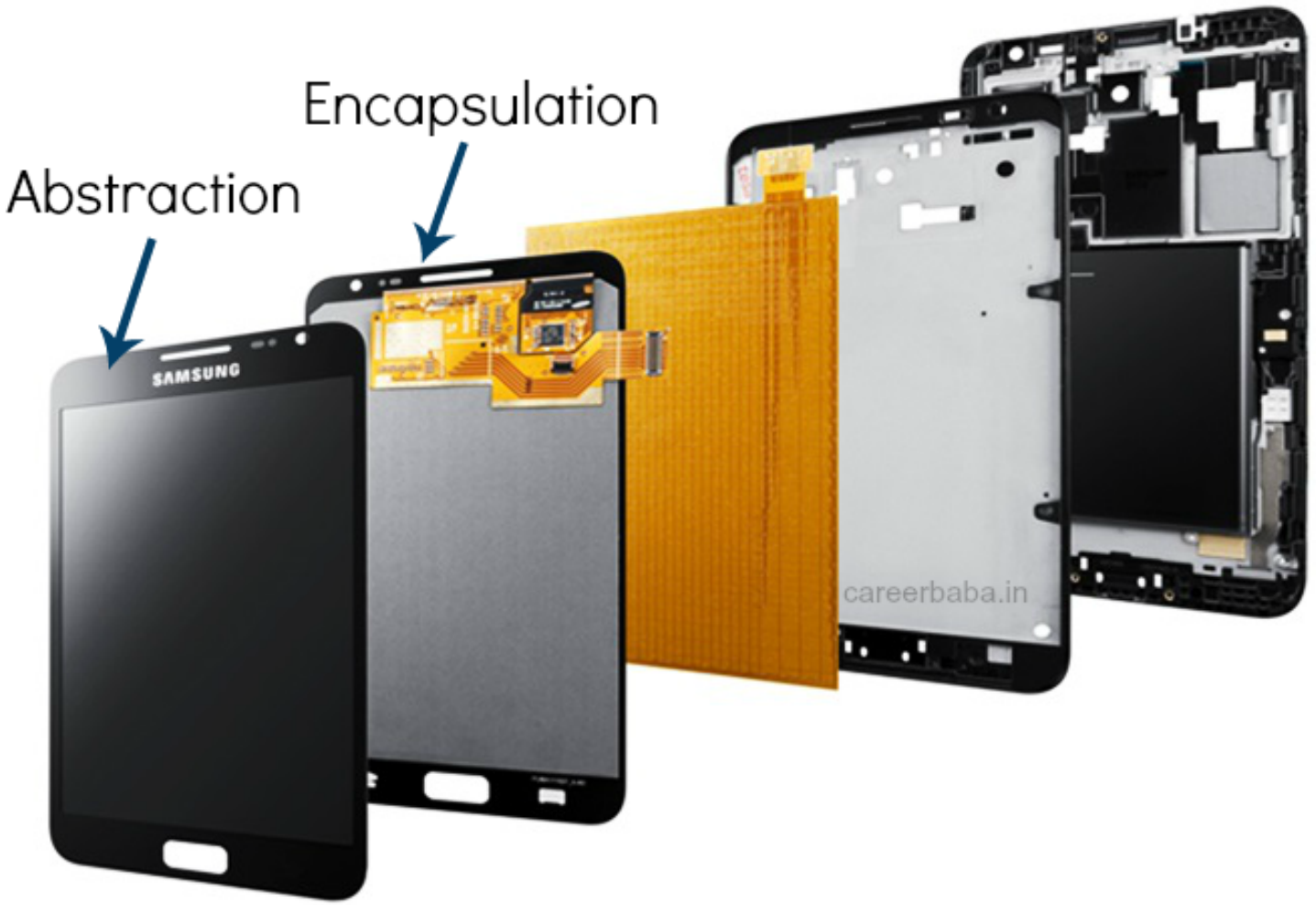
- Complete a task following structured steps based on the interaction of different **objects**:
  - **Objects** are **instances** of a **class**
  - What is a Cosmological Model on a OOPP?
    - **CLASS!!**
- A competent object-oriented programmer should easily identify the **classes** involved on the task, and for each class identify **methods** and **attributes** needed.
  - This is the main point of **abstraction** on OOP!

Object-Oriented Programing Paradigm (2)

- Object
  - Basic unit of object oriented programming. That is both data and function that operate on data.
- Class
  - Blueprint for an object.
- Abstraction
  - The essential information to the outside world.
- Encapsulation
  - Placing the data and the functions that work on that data in the same place.
- Inheritance
  - Forming a new class from an old one. The existing class is the **base**. New class is the **derived**.
- Polymorphism
  - The ability to use an operator or function in different ways.

Highlighted from https://www.tutorialspoint.com

Object-Oriented Programing Paradigm (3)



Classes (1)

- Created with the “class” statement with its content being limited by indentation
- Pythonic classes have some special **methods:**
  - They are usually delimited by double underscores
  - The “__init__” method is the method that is executed at class’s instantialization.
  - All class’s methods must have an additional argument declaration at **first** position; usually the word “self” is chosen.
  - Apart from the mandatory “self” parameter, the other parameters of “__init__” define which attributes should be passed in order to instantialize the class

Coding…
(hello.py)

Classes (2)

- Class/Object Variables
  - Inside a class you can define variables that are exclusive to the object or shared by the whole class
    - Inside a class *self.var* defines *var* as an object variable
    - Inside a class *var* defines *var* as an class variable
- Private variables
  - If a variable begins with “_” it is a convention that it is intended to be private
  - If a variable begins with “__” Python uses name-mangling to make it private

Coding…
(robots.py)

Classes (3)

- Encapsulation:
  - Encapsulation is one of the key elements of any OOP language; setters and getters are the proper way of accessing and modifying an attribute
  - Properties are special attributes of which you have complete control on his getter, setter and deleter
  - @properties are the Pythonic way of defining setters and getters

Coding…
(cosmology.py)

Classes (4)

- Inheritance:
  - Forming a new class from an existing class
  - A competent OOP programmer should identify “master” and inherited classes and their relationship

Coding…
(school.py)

Classes (5)

- Polymorphism and Overloading:
  - The ability to use an operator or function in different ways
  - Polymorphism prevents creating different functions for the same purpose but with different attributes
  - Overloading enhances the readability of the program, adding a new functionality for an old operator

Coding…
(Polymorphism)

Challenge (1)

Think about a Cosmological Model Class:

- What could be the __init__ method?
- What other methods should it contain?
- What could be an example of encapsulation?
- What could be a possible inherited class?
- What could be a possible polymorphic method?

Challenge (2)

Write an iterator to compute the Fibonacci series:

- An iterator in Python must have two methods: __iter__ and next;
  - __iter__ returns object it**self**
  - next returns the next element

Challenge (3)

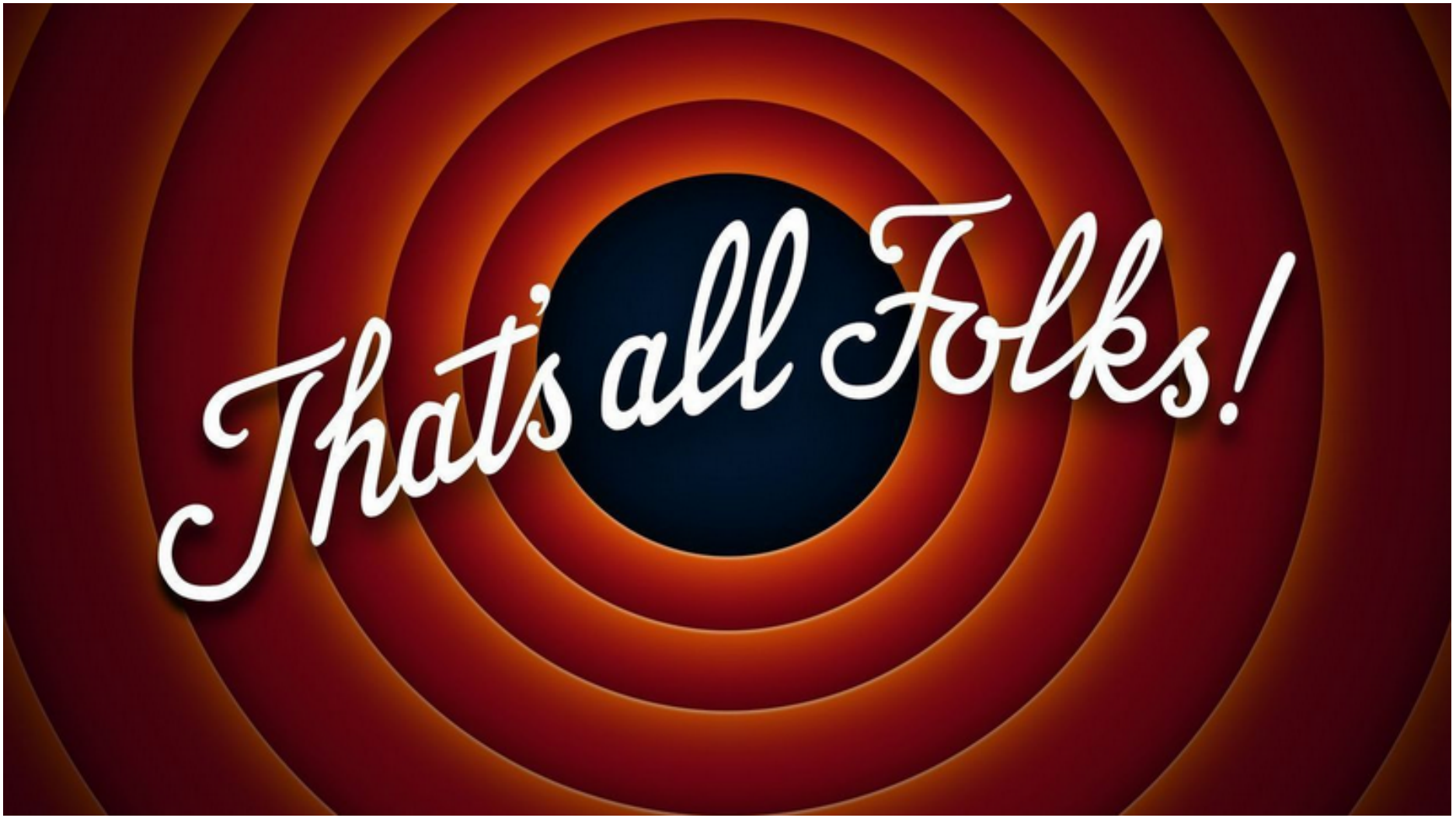
Write a simplistic version of a numeric array, which:

- The operators +,-,*,/ are overloaded to operate element wise

Homework:

Write a cosmological model class:

- Basic Functionalities: Distance Calculator and Power Spectrum Computation
- Advanced Functionalities: Create an inherited class that implements the Press–Schechter formalism


That's all Folks!