

# Εισαγωγή στη γλώσσα προγραμματισμού Python

**Note:** This notebook is heavily influenced by the lectures of Dr. Thomas Erben @ University of Bonn

Τα τελευταία χρόνια παρατηρείται μία ραγδαία αύξηση στον όγκο των δεδομένων που δημιουργούνται σε καθημερινή βάση, τόσο στους κόλπους της ακαδημαϊκής κοινότητας όσο και στον ιδιωτικό τομέα. Η επεξεργασία και ανάλυση των δεδομένων για την επίτευξη των στόχων μας, έχει δημιουργήσει μία νέα επιστήμη που έχει πλέον παγιώσει τη θέση της σε παγκόσμιο επίπεδο και καθιστά την κατανόηση βασικών εννοιών της αναγκαία.

## Το επιστημονικό οικοσύστημα

Στη σημερινή εποχή, ένας ερευνητής φυσικός βασίζεται, περισσότερο από κάθε άλλη φορά, σε υπολογιστικά εργαλεία για να φέρει εις πέρας την έρευνά του.

Από τη θεωρητική φυσική έως την πειραματική και παρατηρησιακή φυσική, η ανάπτυξη της επιστήμης βασίζεται στην εργαλειοποίηση αριθμητικών και υπολογιστικών μεθόδων για προσομοιώσεις φυσικών φαινομένων, αποθήκευση και ανάκληση τεράστιου όγκου δεδομένων, επεξεργασία εικόνων/σημάτων κτλ.

Για αυτό το λόγο, είναι μείζονος σημασίας ο φοιτητής να εκτεθεί και να κατανοήσει τους τρόπους με τους οποίους μπορούμε να αξιοποιήσουμε την υπολογιστική ισχύ που έχουμε διαθέσιμη.

Στο πεδίο της φυσικής, τα πιο συχνά υπολογιστικά εργαλεία που θα κληθεί να χρησιμοποιήσει κάποιος είναι:

### 1. Γλώσσες προγραμματισμού χαμηλού επιπέδου (C/C++, Fortran)

- Είναι οι πιο παλιές και πιο διαδεδομένες γλώσσες προγραμματισμού στον κόσμο της Φυσικής.
- Είναι κατασκευασμένες με γνώμονα τη βέλτιστη απόδοση και όχι με γνώμονα τη φιλικότητα ως προς τον χρήστη.
- Όντας από τις πιο παλιές γλώσσες, υπάρχουν εκατοντάδες βιβλιοθήκες που έχουν βελτιστοποιηθεί για την επίλυση συγκεκριμένων επιστημονικών προβλημάτων → δεν μπορούμε να αποφύγουμε τη χρήση τους.
- Ο υπολογιστικός χρόνος είναι πολυτιμότερος από τον ανθρώπινο χρόνο!
- Αυτές οι γλώσσες προγραμματισμού δεν προσφέρουν άμεση πρόσβαση σε υψηλού-επιπέδου εργαλεία ανάλυσης δεδομένων → πρέπει να καταφύγουμε σε άλλο λογισμικό (π.χ. οπτικοποίηση με `gnuplot`).
- Ο υπάρχων κώδικας που θα κληθείτε να αξιοποιήσετε πολλές φορές είναι κακογραμμένος (κακή τεκμηρίωση, πολύ δυσανάγνωστος).
- Το πιο πιθανό είναι να χρειαστεί να δουλέψετε με κάποια από αυτές ανάλογα με το ερευνητικό γκρουπ στο οποίο θα είστε.

### 2. Διαδραστικές γλώσσες προγραμματισμού (Mathematica, MATLAB, IDL)

- Διαδραστική οπτικοποίηση.
- Εκτεταμένες βιβλιοθήκες (IDL `astro package`).

- Δύσκολες να χρησιμοποιηθούν εκτός διαδραστικού περιβάλλοντος ή για μεγάλο όγκο δεδομένων.
- Έχουν βελτιστοποιηθεί για υλοποίηση μικρού ή μεσαίου μεγέθους έργων.
- **Πολύ ακριβό, ιδιόκτητο software** → πολύ σπάνια θα βρείτε κάποιον φορέα/ινστιτούτο που είναι διατεθειμένος να ξοδέψει πόρους για διάθεση ιδιόκτητου software όταν υπάρχουν άλλα open source εργαλεία → Υπάρχει κίνδυνος να μην είστε σε θέση να αξιοποιήσετε κώδικα που έχετε γράψει παλιότερα (όταν σταματήσετε να έχετε την φοιτητική σας ιδιότητα).
- Συνήθως είναι υπερεξειδικευμένες.

### 3. Γραμμή εντολών (τερματικό) Unix

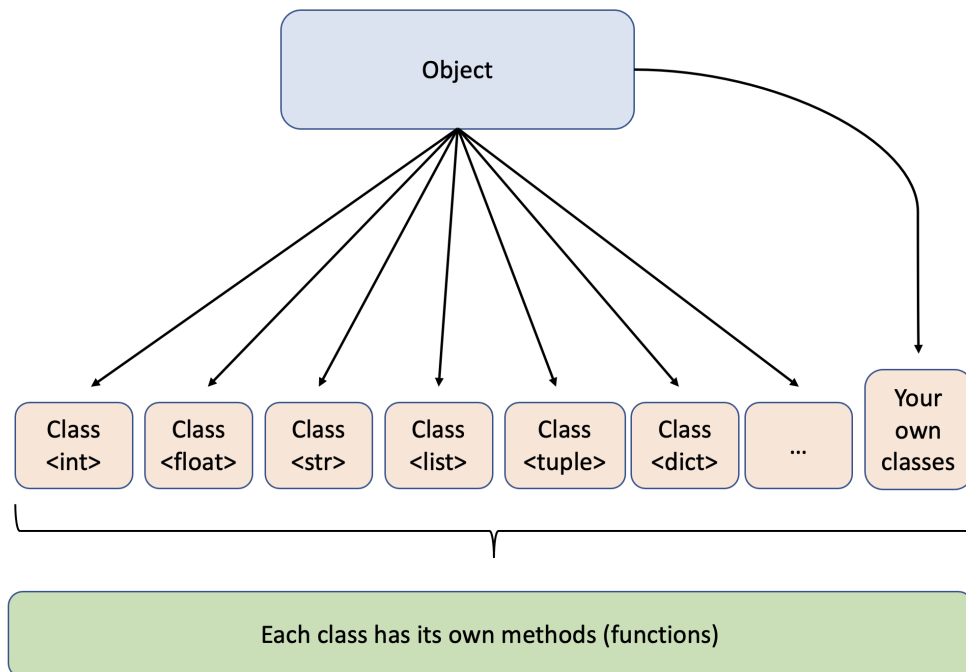
- Απαιτεί γνώση πολλών διαφορετικών εργαλείων και γλωσσών προγραμματισμού (grep, bash, gnuplot, ...)

## Ποιά είναι η θέση της Python σε αυτό το οικοσύστημα;

Γιατί πρέπει κάποιος να επενδύσει χρόνο ώστε να μάθει (και) Python;

### Πλεονεκτήματα της Python

- Η Python είναι μία υψηλού επιπέδου, μη-μεταγλωττίσιμη, γενικής-χρήσης διερμηνεύσιμη γλώσσα προγραμματισμού που προσφέρει πολλές ισχυρές βιβλιοθήκες (modules) για επιστημονική χρήση.
- Εκτός των βιβλιοθηκών, προσφέρει μία πλούσια συλλογή από ενσωματωμένες (built-in) δομές δεδομένων (λίστες, σύνολα, λεξικά κτλ).
- Έχει ένα πολύ εύκολο και ευανάγνωστο συντακτικό. Πολλές φορές μοιάζει με ψευδοκώδικα.
- Η απλότητα και αμεσότητα του συντακτικού της την καθιστά ιδανική επιλογή για πρώτη γλώσσα προγραμματισμού.
- Ακόμα και μη-επαγγελματίες μπορούν σε σύντομο χρονικό διάστημα να αξιοποιούν τις δυνατότητες που προσφέρει σε ικανοποιητικό βαθμό.
- Η Python δεν περιορίζεται στη χρήση αριθμητικών εννοιών. Μπορούμε να τη χρησιμοποιήσουμε για αλληλεπίδραση με ιστοτόπους, βάσεις δεδομένων κτλ, ανάλυση και οπτικοποίηση δεδομένων, ανάπτυξη ιστοσελίδων, τεχνητή νοημοσύνη και πολλά άλλα.
- Τα πάντα στην Python είναι αντικείμενα. Προσφέρει τη δυνατότητα αντικειμενοστραφούς προγραμματισμού αλλά δεν είναι απαραίτητο να δουλεύετε με κλάσεις (σε αντίθεση με αμιγώς αντικειμενοστραφείς γλώσσες όπως η Java).



- Μπορεί να συνδυαστεί με τις γλώσσες C και Fortran (δες Cython).
- Είναι εξαιρετικά δημοφιλής γλώσσα τόσο στην ακαδημία όσο και στον ιδιωτικό τομέα.
- Με την Python μπορείτε να καλύψετε περισσότερο από το 90% των αναγκών σας.

## Μειονεκτήματα της Python

- Η Python είναι *μέχρι* και μία τάξη μεγέθους πιο αργή από τη C και τη Fortran. Έχουμε πολύ λιγότερο έλεγχο της μνήμης.
- Υπάρχουν τόσα πολλά διαθέσιμα εργαλεία που δεν υπάρχει ένα ξεκάθαρο σημείο έναρξης για αρχάριους.
- Υπάρχουν δύο εκδόσεις που δεν είναι συμβατές μεταξύ τους (δες Python2 vs. Python3).

## Παραδείγματα εφαρμογών της Python

### Διαδραστικότητα - Ανάλυση και οπτικοποίηση δεδομένων

Σε αυτό το μέρος, θα δούμε μία απλή και διαδραστική ροή εργασίας με την Python.

Αρχικά εξετάζουμε ένα απλό σύνολο δεδομένων που περιέχει δύο στήλες. Για να το επιτύχουμε αυτό χρησιμοποιούμε το κελί όπως θα χρησιμοποιούσαμε το τερματικό Unix, με το συντακτικό `!some_command`. Οι δύο στήλες στο αρχείο έχουν το ακόλουθο νόημα:

1. Κάποια ημερομηνία. Το δεκαδικό μέρος του αριθμού μας δίνει τη μέρα του έτους. Για παράδειγμα, η 1η Ιανουαρίου δίνεται ως  $1/365 = 0.00274$ . Τα δισεκτα έτη μπορούν να αγνοηθούν.
2. Τη μέση θερμοκρασία της συγκεκριμένης ημέρας στη πόλη του Μονάχου (Γερμανία).

Στη συνέχεια, κατασκευάζουμε ένα γράφημα των δεδομένων και απορρίπτουμε έκτοπες τιμές (θα συζητήσουμε αναλυτικά για την ανάλυση και οπτικοποίηση δεδομένων σε μεταγενέστερο χρόνο).

In [1]:

```
# list the first 5 entries of the file 'data/munich_temperatures.txt'
# with the Unix head command
!head -5 data/munich_temperatures.txt
```

```
1995.00274 0.944444
1995.00548 -1.61111
1995.00821 -3.55556
1995.01095 -9.83333
1995.01369 -10.2222
```

In [2]:

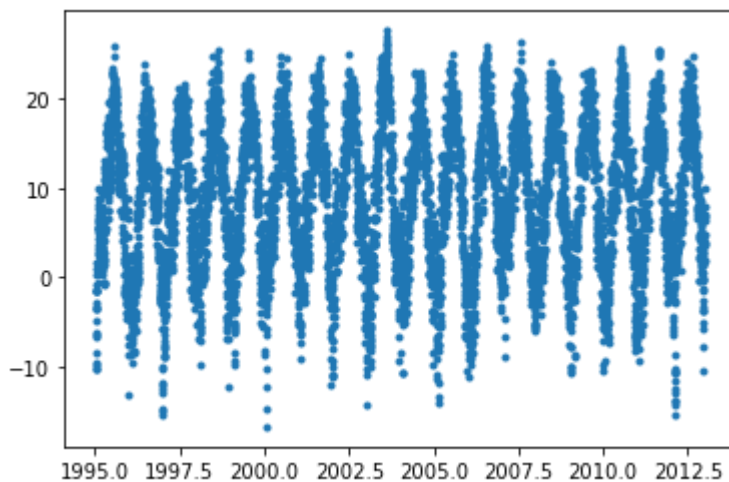
```
# playing around with the data interactively

# The following line is necessary to show matplotlib
# plots within this notebook
%matplotlib inline
import matplotlib.pyplot as plt
import numpy as np

data = np.loadtxt("data/munich_temperatures.txt")
day = data[:,0]
temp = data[:,1]

# get rid of obviously bad data
day = day[(temp < 50) & (temp > -50)]
temp = temp[(temp < 50) & (temp > -50)]

_ = plt.plot(day, temp, '.')
```



## Διαδραστικότητα - Συμβολική επεξεργασία μαθηματικών εκφράσεων

Η βιβλιοθήκη `sympy` μας επιτρέπει να λύσουμε συμβολικά κάποια παράσταση, χωρίς να ασχοληθούμε με τις αριθμητικές ποσότητες που αναπαριστώνται από αυτά τα σύμβολα.

In [3]:

```
import sympy as sp

sp.init_printing()

# declare symbolic variables and functions:
x, y, z, t = sp.symbols('x y z t')
f = sp.Function('f')
```

- Συμβολική αναπαράσταση ενός ολοκληρώματος και η επίλυσή του.

In [4]:

```
sym_integral = sp.Integral(x**2 * sp.cos(x), x)
sym_integral
```

Out[4]:

$$\int x^2 \cos(x) dx$$

In [5]:

```
sym_integral.doit()
```

Out[5]:

$$x^2 \sin(x) + 2x \cos(x) - 2 \sin(x)$$

- Συμβολική αναπαράσταση μιας διαφορικής εξίσωσης και η επίλυσή της.

In [6]:

```
diff_eq = sp.Eq(sp.Derivative(f(x), x, x) + 9*f(x), 1)
diff_eq
```

Out[6]:

$$9 f(x) + \frac{d^2}{dx^2} f(x) = 1$$

In [7]:

```
sp.dsolve(diff_eq, f(x))
```

Out[7]:

$$f(x) = C_1 \sin(3x) + C_2 \cos(3x) + \frac{1}{9}$$

- Συμβολική αναπαράσταση ενός τετραγωνικού πίνακα και εύρεση του αντίστροφου πίνακα.

In [8]:

```
A = sp.Matrix(2, 2, [x, y, z, t])  
A
```

Out[8]:

$$\begin{pmatrix} x & y \\ z & t \end{pmatrix}$$

In [9]:

```
A.inv()
```

Out[9]:

$$\begin{pmatrix} \frac{t}{t x - y z} & -\frac{y}{t x - y z} \\ -\frac{z}{t x - y z} & \frac{x}{t x - y z} \end{pmatrix}$$

## Διαδραστικότητα - Ενσωμάτωση στοιχείων από το διαδίκτυο

Μπορούμε να ενσωματώσουμε στοιχεία από κάποια ιστοσελίδα μέσα στο σημειωματάριο.

In [10]:

```
from IPython.display import YouTubeVideo  
YouTubeVideo("ogvxD9hkb6U")
```

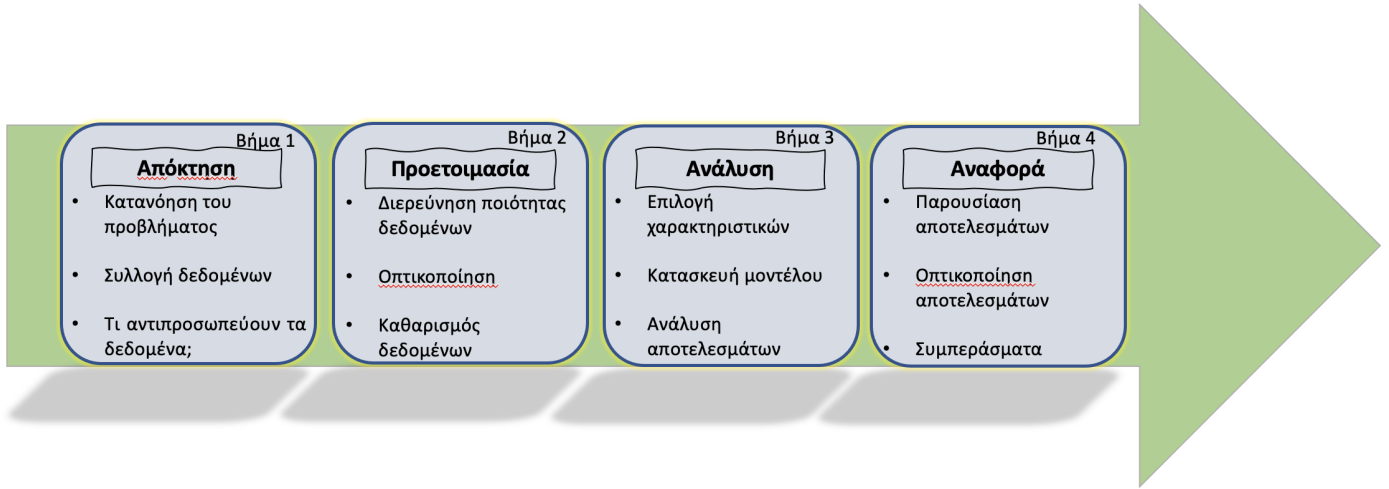
Out[10]:

Department of Physics, University of Cr...



## Στόχοι του μαθήματος

Άμεσος στόχος είναι η εξοικίωση με μια ρουτίνα εργασίας για την επεξεργασία και ανάλυση δεδομένων



Αλλά: Απώτερος στόχος είναι να μάθουμε να αξιοποιούμε υπολογιστικά εργαλεία για να είμαστε σε θέση να αντιμετωπίσουμε μια πληθώρα επιστημονικών προβλημάτων!

### Παράδειγμα:

Η παρακάτω αστρονομική εικόνα αποκαλύπτει κάποιες προβληματικές περιοχές τις οποίες θα θέλαμε να αποκλείσουμε από περαιτέρω ανάλυση. Παλαιότερα, κάτι τέτοιο θα έπρεπε να πραγματοποιηθεί από κάποιον χειροκίνητα αφιερώνοντας πολλές ώρες για να βρει το βέλτιστο τρόπο αποκλεισμού με τη μικρότερη δυνατή απώλεια δεδομένων.

Στη σύγχρονη εποχή, για τον ίδιο σκοπό, θα θέλαμε να αναπτύξουμε ένα πρόγραμμα το οποίο, καταρχάς, θα αναγνωρίζει αυτές τις περιοχές και κατά δεύτερον να εφαρμόζει μία "μάσκα" ώστε να τις καλύπτει αυτόματα. Ένα τέτοιο πρόβλημα θα μπορούσε να είναι μέρος της πτυχιακής/διπλωματικής/διδακτορικής σας διατριβής.



Figure 1: Optical astronomical image

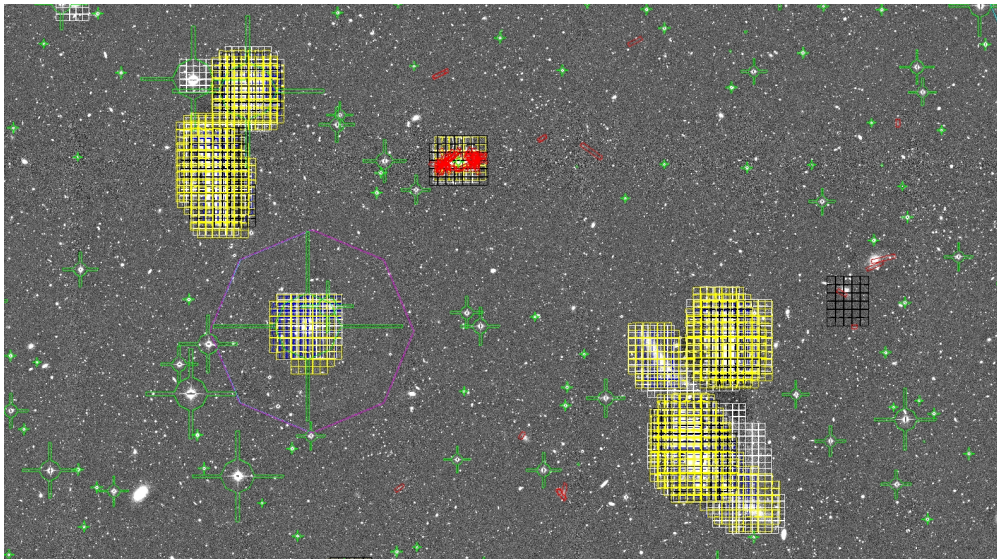


Figure 2: Optical image with automatically generated masks

## Τι πρέπει να γνωρίζετε

- Μία (γενικής χρήσης) γλώσσα προγραμματισμού, τα δομικά της στοιχεία (δομές δεδομένων) και το συντακτικό της. Όλες οι γλώσσες προγραμματισμού γενικής χρήσης μπορούν να λύσουν την ίδια κατηγορία προβλημάτων.
- Για πολύπλοκα προβλήματα χρειάζεται συγκεκριμένες τεχνικές και εργαλεία (προσομοιώσεις, διαφορικές εξισώσεις, αριθμητική ανάλυση)
- Πρέπει να αναπτύξετε αλγοριθμικό τρόπο σκέψης: πως μπορώ να αποδομήσω ένα συγκεκριμένο πρόβλημα ώστε να το λύσω χρησιμοποιώντας τις δυνατότητες μιας γλώσσας προγραμματισμού.
- Πρέπει να αναπτύξετε την ικανότητά σας να ελέγχετε το πρόγραμμά σας για τυχόν σφάλματα ή ειδικές περιπτώσεις που η λειτουργία του δεν είναι πλέον η αναμενόμενη. Πρέπει να είστε σε θέση να έχετε μία αίσθηση για πιθανά προβλήματα που μπορεί να αντιμετωπίσει ο κώδικάς σας.

Φυσικά είναι αδύνατον να καλύψουμε όλα τα παραπάνω μέσα σε έξι εβδομαδιαίες διαλέξεις. Αυτές οι δεξιότητες απαιτούν χρόνο και ενασχόληση ώστε να αρχίσουν να εντυπώνονται στο τρόπο που δουλεύετε.

Σκοπός αυτού του "εντατικού μαθήματος" είναι να σας δώσει κάποιες πρακτικές βάσεις καθώς και το έναυσμα ώστε να αρχίσετε να χρησιμοποιείτε τη γλώσσα προγραμματισμού Python στην καθημερινότητά σας.

Ως προπτυχιακοί φοιτητές/ριες θα έχετε πολλές ευκαιρίες να αναπτύξετε τις προγραμματιστικές σας δεξιότητες μέσω της Python (π.χ. εργαστήρια, πρακτική άσκηση, πτυχιακή κτλ).

**Μην ξεχνάτε:** Μία γλώσσα προγραμματισμού είναι μία γλώσσα που πρέπει να την "μιλάτε" ώστε να βελτιωθείτε!

## Βιβλιογραφία

Το διαδίκτυο και η επίσημη σελίδα της Python είναι ίσως τα πρώτα πράγματα που πρέπει να επισκέφτεστε όταν αντιμετωπίζετε κάποιο πρόβλημα ή ψάχνετε την τεκμηρίωση κάποιας δομής.

Για την εμπέδωση στις έννοιες που θα συζητήσουμε σε αυτό το μάθημα θα πρέπει πολύ συχνά να καταφεύγετε σε επιπρόσθετες πηγές και να πειραματίζεστε μόνοι/ες σας. Η έμπρακτη προσπάθεια να γράψετε κώδικα είναι ο μοναδικός τρόπος να μάθετε μία γλώσσα.



Μην αποθαρρύνεστε αν αρχικά δυσκολεύεστε ή δεν αποδίδετε στο βαθμό που επιθυμείτε. Η αποτυχία είναι *απαραίτητο* στάδιο της εκμάθησης!

Παρακάτω παραθέτουμε ενδεικτικά κάποιες πηγές (στα ελληνικά) που μπορεί να σας φανούν χρήσιμες. Προφανώς η ξένη βιβλιογραφία (έντυπη και διαδικτυακή) είναι πολύ πιο πλούσια. Αξίζει να αφιερώσετε χρόνο και στην αναζήτηση ξενόγλωσσης βιβλιογραφίας καθώς αυτό θα σας επιτρέψει να εξοικιωθείτε και με την ορολογία που χρησιμοποιείται σε παγκόσμιο επίπεδο.

- Η επίσημη ιστοσελίδα της Python: [Python webpage \(https://www.python.org/\)](https://www.python.org/).
- Η ιστοσελίδα [pytoLearn \(http://pytolearn.csd.auth.gr/index.html\)](http://pytolearn.csd.auth.gr/index.html) παρουσιάζει θεμελιώδεις έννοιες που πρέπει να γνωρίζει κανείς ξεκινώντας να προγραμματίζει με την Python.
- Το Κέντρο Ανοικτών Διαδικτυακών Μαθημάτων **mathesis** των Π.Ε.Κ. προσφέρει δωρεάν Μαζικά Ανοικτά Διαδικτυακά Μαθήματα (MOOC). Ένα από αυτά αφορά και την εισαγωγή στην Python. Μπορείτε να βρείτε το μάθημα και να το παρακολουθήσετε σε καθεστώς αυτομελέτης [εδώ \(https://mathesis.cup.gr/courses/course-v1:ComputerScience+CS1.1+2017\\_T2/course/\)](https://mathesis.cup.gr/courses/course-v1:ComputerScience+CS1.1+2017_T2/course/).
- Ν. Α. Αγγελιδάκης, (2019), *Εισαγωγή στον προγραμματισμό με την Python*, Σημειώσεις μαθήματος. Το συγκεκριμένο βιβλίο διατίθεται ελεύθερα στο διαδίκτυο σε ηλεκτρονική μορφή: [Python notes \(http://aggelid.mysch.gr/pythonbook/\)](http://aggelid.mysch.gr/pythonbook/).
- Δ. Λεβεντέας, (2019), *Οδηγός Python Μέσω Παραδειγμάτων*, Ομάδα TasPython. Ο συγκεκριμένος οδηγός διατίθεται ελεύθερα στο διαδίκτυο σε ηλεκτρονική μορφή: [Python guide with examples \(http://python.org.gr/phocadownload/Tutorials/tutorial\\_by\\_example.pdf\)](http://python.org.gr/phocadownload/Tutorials/tutorial_by_example.pdf).