



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΔΙΑΤΜΗΜΑΤΙΚΟ ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΙΚΗΣ ΒΙΟΙΑΤΡΙΚΗΣ**

**ΚΑΤΕΥΘΥΝΣΗ
«ΠΛΗΡΟΦΟΡΙΚΗ ΜΕ ΕΦΑΡΜΟΓΕΣ ΣΤΗΝ ΑΣΦΑΛΕΙΑ,
ΔΙΑΧΕΙΡΙΣΗ ΜΕΓΑΛΟΥ ΟΓΚΟΥ ΔΕΔΟΜΕΝΩΝ ΚΑΙ
ΠΡΟΣΟΜΟΙΩΣΗ»**

**«ΧΡΗΣΗ ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΩΝ ΕΡΓΑΛΕΙΩΝ ΣΤΗΝ
ΔΙΑΘΕΜΑΤΙΚΗ ΠΡΟΣΕΓΓΙΣΗ ΤΗΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΣΤΗΝ
ΦΥΣΙΚΗ ΣΤΗΝ ΔΕΥΤΕΡΟΒΑΘΜΙΑ ΕΚΠΑΙΔΕΥΣΗ»**

Γεώργιος Αποστολόπουλος

ΜΕΤΑΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

**Επιβλέπων
Γεώργιος Σταμούλης**

Λαμία, 2019



UNIVERSITY OF THESSALY
SCHOOL OF SCIENCE
INFORMATICS AND COMPUTATIONAL BIOMEDICINE

**«IMPLEMENTATION OF PROGRAMMING TOOLS IN CROSS-
THEMATIC APPROACH OF INFORMATICS IN PHYSICS IN
SECONDARY EDUCATION»**

George Apostolopoulos

MASTER THESIS

Supervisor
George Stamoulis

Lamia, 2019

«Υπεύθυνη Δήλωση μη λογοκλοπής και ανάληψης προσωπικής ευθύνης»

Με πλήρη επίγνωση των συνεπειών του νόμου περί πνευματικών δικαιωμάτων, και γνωρίζοντας τις συνέπειες της λογοκλοπής, δηλώνω υπεύθυνα και ενυπογράφως ότι η παρούσα εργασία με τίτλο *«Χρήση προγραμματιστικών εργαλείων στην διαθεματική προσέγγιση της Πληροφορικής στην Φυσική στην Δευτεροβάθμια Εκπαίδευση»* αποτελεί προϊόν αυστηρά προσωπικής εργασίας και όλες οι πηγές από τις οποίες χρησιμοποίησα δεδομένα, ιδέες, φράσεις, προτάσεις ή λέξεις, είτε επακριβώς (όπως υπάρχουν στο πρωτότυπο ή μεταφρασμένες) είτε με παράφραση, έχουν δηλωθεί κατάλληλα και ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή. Αναλαμβάνω πλήρως, ατομικά και προσωπικά, όλες τις νομικές και διοικητικές συνέπειες που δύναται να προκύψουν στην περίπτωση κατά την οποία αποδειχθεί, διαχρονικά, ότι η εργασία αυτή ή τμήμα της δεν μου ανήκει διότι είναι προϊόν λογοκλοπής.

Ο ΔΗΛΩΝ
Ημερομηνία

Υπογραφή

**«ΧΡΗΣΗ ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΩΝ ΕΡΓΑΛΕΙΩΝ ΣΤΗΝ
ΔΙΑΘΕΜΑΤΙΚΗ ΠΡΟΣΕΓΓΙΣΗ ΤΗΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΣΤΗΝ
ΦΥΣΙΚΗ ΣΤΗΝ ΔΕΥΤΕΡΟΒΑΘΜΙΑ ΕΚΠΑΙΔΕΥΣΗ»**

Γεώργιος Αποστολόπουλος

Τριμελής Επιτροπή:

Γεώργιος Σταμούλης

Μαρία Κοζύρη

Αθανάσιος Λουκόπουλος

Επιστημονικός Σύμβουλος: Αντώνιος Δαδαλιάρης

ΕΥΧΑΡΙΣΤΙΕΣ

Με την παρούσα μεταπτυχιακή εργασία έρχεται το τέλος μιας όμορφης διαδρομής προς την ολοκλήρωση του μεταπτυχιακού προγράμματος «Πληροφορική με εφαρμογές στην Ασφάλεια, Διαχείριση Μεγάλου Όγκου Δεδομένων και Προσομοίωση» του Διατμητικού Προγράμματος μεταπτυχιακών Σπουδών Πληροφορικής και Υπολογιστικής Βιοϊατρικής του Πανεπιστημίου Θεσσαλίας.

Θα ήθελα να εκφράσω τις θερμές μου ευχαριστίες:

Στον κ. Αντώνιο Δαδαλιάρη για την πολύτιμη καθοδήγησή του στην ανάπτυξη της εργασίας με τις χρήσιμες συμβουλές του.

Τον φίλο και αδελφό μου, Δημήτριο Κατσαδούρο, Φυσικό και καθηγητή Πληροφορικής Μέσης Εκπαίδευσης, για την αμέριστη στήριξη και βοήθειά του καθ' όλη την διάρκεια των μεταπτυχιακών μου σπουδών.

Τέλος, ιδιαίτερα θα ήθελα να ευχαριστήσω, αλλά και να αφιερώσω όλη την προσπάθειά μου για το μεταπτυχιακό πρόγραμμα, στην οικογένειά μου, στη σύζυγό μου για την ηθική στήριξη και στις δύο κόρες μου Αθανασία και Κρυσταλλία που αποτελούν πηγή έμπνευσης και δημιουργικότητας.

Περιεχόμενα

ΠΕΡΙΛΗΨΗ.....	13
A. ΔΙΑΘΕΜΑΤΙΚΟΤΗΤΑ ΣΤΗΝ ΕΚΠΑΙΔΕΥΣΗ.....	15
B. ΡΥΤΗΘΝ.....	17
B.1. Ιστορία και εκδόσεις.....	17
B.2. Γιατί Python?.....	18
B.3. Χαρακτηριστικά.....	19
B.4. Εγκατάσταση Python.....	20
B.5. Modules - Packages.....	20
B.5.1. <i>Module math</i>	23
B.5.2. <i>Package matplotlib</i>	25
B.5.3. <i>Package numpy</i>	29
Γ. ΡΥCHARM - IDE.....	31
Δ. GIT και GITHUB.....	33
E. ΑΠΟΘΕΤΗΡΙΟ.....	35
E.1. Γραφική διεπαφή χρήστη (GUI).....	35
E.2. HTML.....	39
ΣΤ. ΦΥΣΙΚΗ ΓΕΝΙΚΗΣ ΠΑΙΔΕΙΑΣ Β ΛΥΚΕΙΟΥ.....	41
ΣΤ.1. Κεφάλαιο 1. ΔΥΝΑΜΕΙΣ ΜΕΤΑΞΥ ΗΛΕΚΤΡΙΚΩΝ ΦΟΡΤΙΩΝ.....	41
ΣΤ.2. Κεφάλαιο 2. ΗΛΕΚΤΡΙΣΜΟΣ: ΣΥΝΕΧΕΣ ΗΛΕΚΤΡΙΚΟ ΡΕΥΜΑ.....	43
ΣΤ.3. Κεφάλαιο 3. ΦΩΣ.....	44
ΣΤ.4. Κεφάλαιο 4. ΑΤΟΜΙΚΑ ΦΑΙΝΟΜΕΝΑ.....	45
Z. ΑΝΑΦΟΡΕΣ- ΒΙΒΛΙΟΓΡΑΦΙΑ.....	46

ΠΕΡΙΛΗΨΗ

Η παρούσα εργασία με τίτλο: “Χρήση προγραμματιστικών εργαλείων στη διαθεματική προσέγγιση της πληροφορικής στην Φυσική της Δευτεροβάθμιας Εκπαίδευσης”, σκοπό έχει να δημιουργήσει ένα ανοιχτό αποθετήριο λύσεων των ασκήσεων του μαθήματος της Φυσικής Β Λυκείου στο Github, στην πλέον δημοφιλή πλατφόρμα φιλοξενίας κώδικα, χρησιμοποιώντας εργαλεία πληροφορικής ανοιχτού κώδικα και συγκεκριμένα Python, HTML5, CSS.

Θα προσθέσουμε έτσι ένα ακόμα εργαλείο στην διαθεματική προσέγγιση της Φυσικής μέσω της Πληροφορικής και αντίστροφα, καθώς η πληροφορική αποτελούσε πάντα και αποτελεί ένα διασκεδαστικό διεπιστημονικό εργαλείο μάθησης που κεντρίζει το ενδιαφέρον των μαθητών.

Χρησιμοποιούμε μια από τις πλέον διαδεδομένες και ευέλικτες γλώσσες υψηλού επιπέδου όπως είναι η Python, που κατατάσσεται στις πρώτες θέσεις προτίμησης των προγραμματιστών παγκοσμίως, ιδίως μεταξύ των χωρών με τις πιο αναπτυσσόμενες οικονομίες.

Λέξεις κλειδιά: Python, Διαθεματικότητα, Φυσική, HTML, CSS, GitHub, Git, Pycharm.

A. ΔΙΑΘΕΜΑΤΙΚΟΤΗΤΑ ΣΤΗΝ ΕΚΠΑΙΔΕΥΣΗ

Η διαθεματική προσέγγιση της γνώσης αποτελεί μια νέα μέθοδο στην μαθητική διδασκαλία. Το 2003 εισάγεται στο Ελληνικό Εκπαιδευτικό Σύστημα ένα νέο Πρόγραμμα Σπουδών το λεγόμενο Διαθεματικό Ενιαίο Πλαίσιο Προγραμμάτων Σπουδών που θεσμοθετήθηκε με το ΦΕΚ 303 και 304/13-3-2003.

Θεμελιώδεις χαρακτηριστικό του νέου προγράμματος είναι η ολιστική προσέγγιση της μάθησης και η υποστήριξη της ενεργού συμμετοχής του μαθητή στις θεματικές και διαθεματικές δραστηριότητες που διοργανώνει ο διδάσκων σύμφωνα με τις τοπικές πολιτιστικές, κοινωνικές και άλλες συνθήκες.

Με τον όρο διαθεματικότητα εννοούμε μια σειρά από εκπαιδευτικές προσεγγίσεις πάνω στην ενιαιοποίηση της σχολικής γνώσης. Διαθεματική διδασκαλία είναι μια θέαση της γνώσης και μια προσέγγιση του Αναλυτικού Προγράμματος που συνειδητά εφαρμόζει μεθοδολογία και γλώσσα από περισσότερους από έναν κλάδους, προκειμένου να εξετάσει ένα κεντρικό θέμα, ζήτημα, πρόβλημα ή εμπειρία [1].

Η πολυδιάσπαση του διδακτικού περιεχομένου σε επιμέρους επιστημονικούς κλάδους, κατακερματίζει την ανθρώπινη σκέψη. Αυτό είχε ως αποτέλεσμα οι μαθητές να παραμένουν αδιάφοροι για τη γνώση αυτής της μορφής, αλλά και ανίκανοι να αξιοποιήσουν νέες μορφές σκέψης και δράσης, καθώς δεν έχουν την ευκαιρία να συλλάβουν τα κοινά σημεία των επιστημών [2].

Αυτό που εισάγεται λοιπόν είναι μια εκπαιδευτική διαδικασία στην οποία χρησιμοποιούνται περισσότερες γνωστικές περιοχές με σκοπό την αύξηση της μάθησης σε κάθε περιοχή. Η πληροφορική έχει να παίξει εδώ ένα σημαντικό ρόλο.

Στην εργασία αυτή θα προσεγγίσουμε την διδασκαλία της Φυσικής της Β' Λυκείου χρησιμοποιώντας εργαλεία πληροφορικής. Η πληροφορική αποτελούσε πάντα και αποτελεί ένα διασκεδαστικό διεπιστημονικό εργαλείο μάθησης που κεντρίζει το ενδιαφέρον των μαθητών.

[1] Heidi Hayes Jacobs, *Interdisciplinary Curriculum: Design and Implementation*, ISBN-0-87120-165-8, 1989.
[2] Μασταγγούρας Ηλίας, *Η διαθεματικότητα στη σχολική γνώση*, ISBN 960-333-310-7, 2002

Μέσα από αυτήν την εργασία θα δημιουργήσουμε ένα ανοιχτό αποθετήριο λύσεων των ασκήσεων του μαθήματος της Φυσικής Β Λυκείου στο Github που αποτελεί ένα web-based hosting service, χρησιμοποιώντας εργαλεία πληροφορικής ανοιχτού κώδικα και συγκεκριμένα HTML5, CSS, Python.

B. PYTHON

B.1. Ιστορία και εκδόσεις

Η Python είναι μια σύγχρονη γλώσσα προγραμματισμού που ανήκει στην κατηγορία των γλωσσών σεναρίου.

Η δημιουργία της γλώσσας ξεκίνησε στα τέλη της δεκαετίας του '80 από τον Ολλανδό Guido von Rossum. Η πρώτη εμφάνιση της γλώσσας έγινε το 1991 και η έκδοση 1.0 κυκλοφόρησε το 1994. Η Python, που πήρε το όνομά της από την ομάδα των βρετανών κωμικών ηθοποιών Monty Python, είναι λογισμικό ανοικτού κώδικα που σήμερα υποστηρίζεται από μια μεγάλη ομάδα εθελοντών και είναι ελεύθερα διαθέσιμη από το Ίδρυμα Python Software Foundation.

Η κύρια έκδοση της γλώσσας είναι γραμμένη σε γλώσσα C. Αυτή τη στιγμή υπάρχουν δύο κύριες ροές εκδόσεων της Python. Η σειρά των εκδόσεων 2.x αποτελεί την παλαιότερη εκδοχή της, με τελευταία στη σειρά την έκδοση με κωδικό αριθμό 2.7.6 που κυκλοφόρησε το Νοέμβριο του 2013. Η σειρά 2.x εξακολουθεί να υποστηρίζεται για λόγους συμβατότητας και υποστήριξης του λογισμικού που έχει γραφτεί ήδη, αλλά σύμφωνα με το Ίδρυμα Python Software δεν πρόκειται να συνεχίσει η εξέλιξή της. Η νέα σειρά 3.x αποτελεί το παρόν και το μέλλον της γλώσσας. Η τελευταία έκδοση της σειράς είναι η 3.7.3 που κυκλοφόρησε τον Μάρτιο του 2019.[3]

Οι εκδόσεις 3.x δεν είναι συμβατές με τις εκδόσεις 2.x. Η απόφαση αυτή λήφθηκε από τον δημιουργό της γλώσσας προκειμένου να γίνουν αλλαγές στον μεταγλωττιστή που θα του εξασφαλίσουν καλύτερη λειτουργικότητα και βελτιώσεις που διαφορετικά δεν θα μπορούσαν να γίνουν.

Η γλώσσα, πέραν της κύριας έκδοσής της, διατίθεται και σε άλλες υλοποιήσεις που είναι γραμμένες σε διάφορες γλώσσες προγραμματισμού και συνεργάζονται εγγενώς με αυτές, χωρίς να αποκλείεται να διαλειτουργούν και με άλλες γλώσσες μέσα από τις βιβλιοθήκες που διαθέτουν:

[3] Python Documentation by Version, 2019. [Online]. Available: <https://www.python.org/doc/versions/>
[Accessed: 2 May 2019]

- Η Jython είναι μια Java - based έκδοση της Python και μπορεί να χρησιμοποιηθεί για να συνεργαστεί με κώδικα Java.
- Η Iron Python είναι γραμμένη σε C# και συνεργάζεται με το περιβάλλον .net της Microsoft.
- Το PyPy είναι ένα πρότζεκτ που υποστηρίχθηκε από την Ευρωπαϊκή Ένωση αλλά και την Google. Ο πυρήνας του είναι γραμμένος σε ένα υποσύνολο της ίδιας της Python.

B.2. Γιατί Python?

Η Python είναι μια απίστευτα αποτελεσματική γλώσσα: τα προγράμματά της κάνουν περισσότερα σε λιγότερες γραμμές κώδικα από ότι απαιτούν πολλές άλλες γλώσσες προγραμματισμού. Η σύνταξη της Python βοηθάει στην δημιουργία “καθαρού” κώδικα, τον οποίος μπορεί κανείς εύκολα να διαβάσει και να αποσφαλματώσει, μπορεί επίσης εύκολα να επεκταθεί συγκριτικά με άλλες γλώσσες.

Η Python χρησιμοποιείται για πολλούς σκοπούς: δημιουργία παιχνιδιών, χτίσιμο εφαρμογών διαδικτύου, επίλυση προβλημάτων και γενικά ανάπτυξη κάθε είδους προγραμματιστικών εργαλείων σε διάφορους κλάδους επιστημών. Η Python χρησιμοποιείται επίσης στο επιστημονικά πεδίο για την ακαδημαϊκή έρευνα και την εφαρμοσμένη εργασία.

Ένας από τους πιο σημαντικούς λόγους που η Python εξελίσσεται ραγδαία είναι λόγω της κοινότητας Python, η οποία περιλαμβάνει μια απίστευτα ποικιλόμορφη και φιλόξενη ομάδα ανθρώπων. Η Κοινότητα είναι απαραίτητη για τους προγραμματιστές, επειδή ο προγραμματισμός δεν είναι μια μοναχική επιδίωξη. Οι περισσότεροι από τους προγραμματιστές, ακόμη και οι πιο έμπειροι, πρέπει να ζητήσουν συμβουλές από άλλους που έχουν ήδη λύσει παρόμοια προβλήματα. Ο ρόλος μιας καλά οργανωμένης διασυνδεδεμένης και υποστηρικτικής κοινότητας είναι κρίσιμος στην βοήθεια εκμάθησης της γλώσσας και επίλυσης προβλημάτων. [4]

[4] Eric Matthes, "Python Crash Course, A Hands-On, Project-Based Introduction to Programming", San Francisco: William Pollock, 2016

B.3. Χαρακτηριστικά

Τα κυριότερα χαρακτηριστικά της Python, που μπορούν να συνοψιστούν στα ακόλουθα:

Ανοικτός κώδικας. Η γλώσσα διατίθεται δωρεάν τόσο για χρήση όσο και σε επίπεδο κώδικα. Όποιος ενδιαφέρεται μπορεί να βρει τον πηγαίο κώδικα όλων των εκδόσεων από τη διεύθυνση <http://python.org/ftp/python>.

Απλή. Ο κώδικας Python είναι πολύ κοντά στην αγγλική γλώσσα γεγονός που εξασφαλίζει υψηλή αναγνωσιμότητα και ευκολία στην εκμάθηση.

Υψηλού επιπέδου. Η Python ανήκει στις λεγόμενες γλώσσες συγγραφής σεναρίων (scripting languages). Αυτό σημαίνει ότι διαθέτει ισχυρές εντολές με τις οποίες ο προγραμματιστής μπορεί να υλοποιεί τον κώδικά του χωρίς να ασχολείται με θέματα όπως είναι η διαχείριση μνήμης, η υπερχείλιση αριθμών, κ.λπ. Με τον τρόπο αυτό, ο προγραμματιστής μπορεί να εστιάζει στο πρόβλημα που έχει να επιλύσει και όχι στις ιδιαιτερότητες της γλώσσας που χρησιμοποιεί.

Τρέχει μέσω Διερμηνευτή. Οι παραδοσιακές γλώσσες προγραμματισμού διαθέτουν μεταγλωττιστή (Compiler) ο οποίος μετατρέπει τον πηγαίο κώδικα σε δυαδικό (γλώσσα μηχανής). Η διαδικασία αυτή της μεταγλώττισης πρέπει να προηγηθεί της εκτέλεσης του προγράμματος. Ο κώδικας Python εκτελείται μέσω διερμηνευτή (Interpreter). Σε αυτό το μοντέλο, ο πηγαίος κώδικας μετατρέπεται σε έναν ενδιάμεσο κώδικα (ψηφιοκώδικας, bytecode) που στη συνέχεια εκτελείται από την εικονική μηχανή της Python (Python Virtual Machine). Το γεγονός αυτό απαλλάσσει τον προγραμματιστή από θέματα που αφορούν στη μεταγλώττιση, στην ύπαρξη των κατάλληλων βιβλιοθηκών κ.λπ., ενώ της προσδίδει φορητότητα.

Φορητή. Η σχεδίαση που χρησιμοποιεί διερμηνευτή και εικονική μηχανή, την καθιστά ικανή να εκτελείται σε διαφορετικές πλατφόρμες (Windows, Linux, Mac, κ.ά.) χωρίς να απαιτείται αλλαγή στον κώδικα.

Αντικειμενοστρεφής αλλά και Διαδικαστική. Ο προγραμματιστής μπορεί να γράψει είτε κώδικα βασισμένο τις αρχές του διαδικαστικού προγραμματισμού ή κώδικα πλήρως αντικειμενοστρεφή. Ο διαδικαστικός (Procedural Programming), όπως και ο συγγενής του δομημένος προγραμματισμός, υποστηρίζεται από τις παραδοσιακές γλώσσες. Τα

δεδομένα και οι διαδικασίες, που επιδρούν σε αυτά, είναι διαχωρισμένα και η επεξεργασία γίνεται μέσα από συναρτήσεις που επιδρούν πάνω στα δεδομένα. Ο αντικειμενοστρεφής προγραμματισμός (Object Oriented Programming) στηρίζει τη λειτουργία του στα αντικείμενα που συνδυάζουν τα δεδομένα με τις λειτουργίες τους σε μια ολότητα, η οποία υλοποιείται με τη μορφή κλάσεων.

Επεκτασιμότητα. Η ίδια η γλώσσα είναι επεκτάσιμη καθώς ένα βασικό σύνολο της γλώσσας αποτελεί τον πυρήνα της, ενώ όλα τα υπόλοιπα είναι αρθρώματα (modules) που επεκτείνουν τη λειτουργικότητά της. Σε συνδυασμό με το ότι είναι ανοικτού κώδικα, την βοηθάει να μην μένει στάσιμη αλλά και να παρακολουθεί τις εξελίξεις. [5]

B.4. Εγκατάσταση Python

Σχεδόν όλες οι διανομές Linux (π.χ. Ubuntu, Debian, Arch Linux) έρχονται με προεγκατεστημένη την έκδοση 3.x της Python. Σε διαφορετική περίπτωση η εγκατάστασή από το terminal γίνεται με την εντολή:

```
sudo apt-get update && sudo apt-get install python3
```

B.5. Modules - Packages

Module είναι ένα αρχείο που περιέχει ορισμούς και δηλώσεις Python. Το όνομα του αρχείου είναι το όνομα του module με το επίθημα .py στο τέλος.

Τα modules επιτρέπουν την λογική οργάνωση κώδικα Python. Ομαδοποίηση σχετικού κώδικα σε ένα module καθιστά τον κώδικα πιο κατανοητό και χρήσιμο. Σε ένα module μπορεί να ορίσει κανείς λειτουργίες, κλάσεις και μεταβλητές, μπορεί επίσης να περιλαμβάνει κώδικα με δυνατότητα εκτέλεσης. Όπως αναφέρθηκε και παραπάνω τα modules επιτρέπουν την επέκταση της γλώσσας.

Είναι δυνατή η εισαγωγή όλων των ονομάτων (συναρτήσεων, δηλώσεων κ.λ.π) από ένα module στον τρέχοντα χώρο ονομάτων χρησιμοποιώντας την ακόλουθη δήλωση εισαγωγής:

[5] Δρ. Νικόλαος Ζάχαρης, Δρ. Γεώργιος Μαυρομάτης, "Εκπαίδευση στην Python", Αθήνα: Εθνικό Κέντρο Δημόσιας Διοίκησης & Αυτοδιοίκησης, 2014

```
#!/usr/bin/python3
```

```
from modname import *
```

Είναι επίσης δυνατή η μεμονωμένη εισαγωγή ονομάτων (συναρτήσεων, δηλώσεων κ.λ.π) χρησιμοποιώντας την ακόλουθη δήλωση εισαγωγής:

```
#!/usr/bin/python3
```

```
from modname import foo
```

όταν γίνεται η εισαγωγή ενός module ο interpreter της Python αναζητά το module αυτό στις παρακάτω διαδρομές:

- Στον τρέχοντα φάκελο
- Αν το module δεν βρεθεί εκεί, τότε ψάχνει σε κάθε φάκελο που είναι δηλωμένο στην PYTHONPATH environment variable
- Διαφορετικά, η Python ψάχνει στο default path, που σε Linux λειτουργικό είναι συνήθως `/usr/local/lib/python3/`

Η ενσωματωμένη συνάρτηση `dir()` μας επιστρέφει μια ταξινομημένη λίστα αλφαριθμητικών που αντιστοιχούν στα ονόματα των μεθόδων και των μεταβλητών που έχουν δηλωθεί στο module, π.χ. αν εκτελέσουμε:

```
#!/usr/bin/python3
```

```
# Import built-in module math
```

```
import math
```

```
content = dir(math)
```

```
print (content)
```

θα πάρουμε:

```
['__doc__', '__file__', '__name__', 'acos', 'asin', 'atan',  
  
'atan2', 'ceil', 'cos', 'cosh', 'degrees', 'e', 'exp',  
  
'fabs', 'floor', 'fmod', 'frexp', 'hypot', 'ldexp', 'log',  
  
'log10', 'modf', 'pi', 'pow', 'radians', 'sin', 'sinh',  
  
'sqrt', 'tan', 'tanh']
```

Τα πακέτα (packages) επιτρέπουν μια ιεραρχική δομή του χώρου ονομάτων των μονάδων χρησιμοποιώντας dot.notation. Με τον ίδιο τρόπο που τα modules βοηθούν στην αποφυγή συγκρούσεων μεταξύ των ονομασιών των μεταβλητών, τα packages συμβάλλουν στην αποφυγή συγκρούσεων μεταξύ των ονομάτων module. Στην ουσία η δημιουργία ενός πακέτου είναι η κατανομή διαφόρων module (αρχείων py) σε ιεραρχικές δομές (φάκελοι) του λειτουργικού συστήματος. Η κλήση τους γίνεται επίσης με την δήλωση:

```
#!/usr/bin/python3  
  
# Import package module  
  
import package.modulename1, package.modulename2
```

Παρακάτω θα δούμε μερικά βασικά εργαλεία (module και packages) της Python που χρησιμοποιήθηκαν στην δημιουργία αυτής της εργασίας, το *math* το *matplotlib* και το *numpy*.

B.5.1. Module math

Το math module χρησιμοποιείται για πρόσβαση σε μαθηματικές συναρτήσεις της Python. Όλες οι μαθηματικές συναρτήσεις που παρέχει το math module χρησιμοποιούνται για ακέραιους ή πραγματικούς αριθμούς και όχι για μιγαδικούς.[6]

Για να χρησιμοποιήσουμε αυτό το module, θα πρέπει πρώτα να το εισάγουμε στον κώδικα μας, αυτό γίνεται με την δήλωση:

```
#!/usr/bin/python3  
  
import math # Import module math
```

επίσης μπορούμε να εισάγουμε συγκεκριμένες συναρτήσεις ή σταθερές π.χ.:

```
#!/usr/bin/python3  
  
from math import sqrt # Import only sqrt from module math  
  
x=sqrt(2)
```

Σταθερές

Οι σταθερές που είναι δηλωμένες στο math είναι:

```
math.pi # Η μαθηματική σταθερά π = 3.141592  
math.e # Η βάση του φυσικού λογαρίθμου e = 2.718281  
math.tau # Η σταθερά ταυ (2*π) tau = 6.283185  
math.inf # Επιτρέπει το άπειρο  
math.nan # Δεν είναι ένας τύπος αριθμού
```

[6] math — Mathematical functions, Python 3.7.3 documentation, 2019. [Online]. Available: <https://docs.python.org/3/library/math.html> [Accessed: 4 May 2019]

Συναρτήσεις

Συνάρτηση	Περιγραφή
Συναρτήσεις αριθμών	
ceil(x)	Επιστρέφει την στρογγυλοποίηση προς τα πάνω του αριθμού x. Επιστρέφει ακέραιο $\geq x$
copysign(x, y)	Επιστρέφει τον αριθμό x και αντιγράφει το πρόσημο του y στο x.
fabs(x)	Επιστρέφει την απόλυτη τιμή του x
factorial(x)	Επιστρέφει το παραγοντικό του x. όπου $x \geq 0$
floor(x)	Επιστρέφει την στρογγυλοποίηση προς τα κάτω του αριθμού x. . Επιστρέφει ακέραιο $\leq x$
fsum(iterable)	Επιστρέφει το άθροισμα των στοιχείων από σε ένα επαναληπτικό αντικείμενο π.χ. list
gcd(x, y)	Επιστρέφει τον μέγιστο κοινό διαιρέτη των x και y
isfinite(x)	Ελέγχει εάν το x δεν είναι ούτε άπειρο ούτε αριθμός
isinf(x)	Ελέγχει αν το x είναι άπειρο
isnan(x)	Ελέγχει εάν το x δεν είναι αριθμός.
remainder(x, y)	Επιστρέφει το υπόλοιπο της διαίρεσης x δια y
Συναρτήσεις δυνάμεων και λογαριθμικές συναρτήσεις	
pow(x, y)	Επιστρέφει το αποτέλεσμα της δύναμης xy
sqrt(x)	Επιστρέφει την τετραγωνική ρίζα του x
exp(x)	Επιστρέφει το αποτέλεσμα e^x
log(x[, base])	Επιστρέφει τον λογάριθμο του x στην δοσμένη βάση. Η

προεπιλεγμένη βάση είναι ο αριθμός e .

$\log_2(x)$ Επιστρέφει τον λογάριθμο του x με βάση το 2.

$\log_{10}(x)$ Επιστρέφει τον λογάριθμο του x με βάση το 10.

Τριγωνομετρικές συναρτήσεις και συναρτήσεις μετατροπής γωνιών

$\sin(x)$ Επιστρέφει το ημίτονο του x σε ακτίνια

$\cos(x)$ Επιστρέφει το συνημίτονο του x σε ακτίνια

$\tan(x)$ Επιστρέφει την εφαπτομένη του x σε ακτίνια

$\text{asin}(x)$ Αντίστροφο του $\sin(x)$

$\text{acos}(x)$ Αντίστροφο του $\cos(x)$

$\text{atan}(x)$ Αντίστροφο του $\tan(x)$

$\text{degrees}(x)$ Μετατρέπει την γωνία x από ακτίνια σε μοίρες

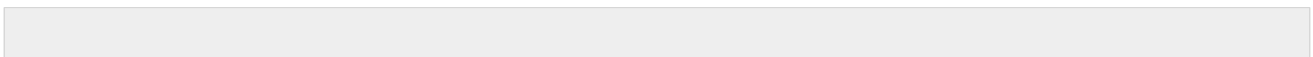
$\text{radians}(x)$ Μετατρέπει την γωνία x από ακτίνια σε μοίρες

B.5.2. Package matplotlib

Ένα από τα πιο δημοφιλή εργαλεία στην Python είναι το `matplotlib`, μια μαθηματική βιβλιοθήκη σχεδίασης. Το `matplotlib` χρησιμοποιείται ευρέως στην Python για Data visualization (Οπτικοποίηση δεδομένων). Μπορεί επίσης να χρησιμοποιηθεί για την γραφική αναπαράσταση οποιουδήποτε dataset. Το `matplotlib` το χρησιμοποιούμε στην άσκηση 2.5 για να αποδώσουμε την γραφική παράσταση ενός αγωγού σε συνάρτηση του μήκους του, της διατομής του, της τάσης στα άκρα του και του ρεύματος που τον διαρρέει.

Η εγκατάσταση του `matplotlib` γίνεται με τις παρακάτω εντολές τερματικού:

είτε:



```
sudo apt-get install python3-matplotlib
```

ή μέσω pip:

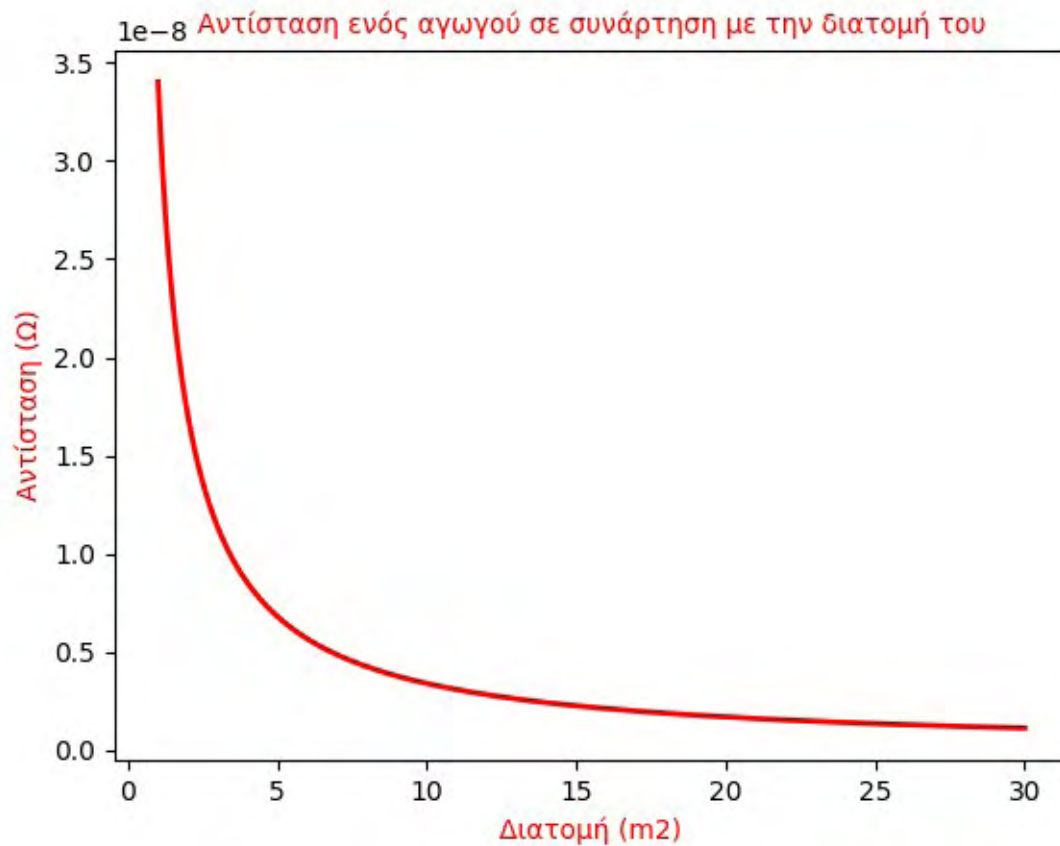
```
pip3 install matplotlib
```

Το PIP είναι ένας διαχειριστής πακέτων (package manager) της Python, το οποίο για την έκδοση Python 3.4 και μετά έρχεται προ εγκατεστημένο σε συστήματα Debian. Αλλιώς η εγκατάσταση του γίνεται με την εντολή

```
sudo apt update && apt install python3-pip
```

Για να δει κάποιος τα είδη απεικονίσεων που μπορεί να κάνει με το matplotlib, αρκεί να επισκεφτεί την ιστοσελίδα <http://matplotlib.org/>.

Με βάση την άσκηση 2.5 μπορούμε να επεξηγήσουμε ορισμένες βασικές μεθόδους της matplotlib.



Σχήμα 1.

```
#!/usr/bin/python3
#filename 2_5.py
"""
Άσκηση 2.5

Να κάνετε τη γραφική παράσταση της αντίστασης ενός αγωγού σε συνάρτηση με:
α) το μήκος του
β) το εμβαδό διατομής τους
γ) την τάση στα άκρα του
δ) την ένταση του ρεύματος που τον διαρρέει.

"""
import matplotlib.pyplot as plt
import numpy as np

def R(l):
    ρ=1.7*10**(-8) #Ειδική αντίσταση χαλκού σε Ωmself.
    S=2*10**(-4) #Διατομή Χαλκού σε m2self
    R=ρ*l/S
    return R
```

```

def RS(S):
    ρ=1.7*10**(-8) #Ειδική αντίσταση χαλκού σε Ωmself.
    l=2           #Μήκος αγωγού σε m
    R=ρ*l/S
    return R

l= np.linspace(0,30,1000)           #Εύρος του l από 0-30 με 1000 τιμές δειγματοληψία
plt.title('Αντίσταση ενός αγωγού σε συνάρτηση με το μήκος του', fontsize=10, color='green') #τίτλος
#γραφήματος
plt.xlabel('Μήκος l (m)',fontsize=10, color='green') # x label
plt.ylabel('Αντίσταση (Ω)',fontsize=10, color='green') # x label
plt.plot(l, Rl(l), 'g-', linewidth=2.0)
plt.show()

S=np.linspace(1,30,1000)           #Εύρος του S από 1-30 με 1000 τιμές δειγματοληψία
plt.title('Αντίσταση ενός αγωγού σε συνάρτηση με την διατομή του', fontsize=10, color='red') #τίτλος
#γραφήματος
plt.xlabel('Διατομή (m2)',fontsize=10, color='red') # x label
plt.ylabel('Αντίσταση (Ω)',fontsize=10, color='red') # x label
plt.plot(S, RS(S), 'r-', linewidth=2.0)
plt.show()

```

είναι προφανές ότι με την δήλωση: `import matplotlib.pyplot as plt` κάνουμε εισαγωγή το module `matplotlib` και του αποδίδουμε το alias: `plt`

Στην συνέχεια με την μέθοδο `title` δημιουργούμε τον τίτλο του γραφήματος. Η μέθοδος `title` παίρνει ορίσματα το μέγεθος της γραμματοσειράς και το χρώμα της.

Η μέθοδος `xlabel` μας επιτρέπει να ορίσουμε το τίτλο του x άξονα, το μέγεθος της γραμματοσειράς και το χρώμα.

Αντίστοιχα για τον y άξονα χρησιμοποιούμε την `ylabel`.

Τέλος με την μέθοδο `plot` - που παίρνει ορίσματα τις τιμές της διατομής του αγωγού στον x άξονα και τις τιμές της αντίστασης στον y άξονα, το χρώμα και το πάχος της γραμμής του γραφήματος- αναπαράγουμε γραφικά την συνάρτηση υπολογισμού της αντίστασης του αγωγού συναρτήσει της διατομής του σχ.1.

Στο σημείο αυτό θα πρέπει να επεξηγήσουμε την χρήση του module `numpy`.

B.5.3. Package numpy

Το numpy αποτελεί ένα από τα πιο βασικά εργαλεία τις Python στην χρήση της στο επιστημονικό πεδίο. Το όνομά του προέρχεται από τις λέξεις Numerical Python και είναι κατά βάση γραμμένο σε C. [7]

Το NumPy εμπλουτίζει την Python με ισχυρές δομές δεδομένων, επιτρέποντας την χρήση πολυδιάστατων πινάκων. Η χρήση του στοχεύει ακόμη και σε τεράστιες δομές πινάκων, που είναι γνωστότερες με την ορολογία *Big Data*. Εκτός αυτού, παρέχει μια μεγάλη βιβλιοθήκη μαθηματικών μεθόδων υψηλού επιπέδου για την χρήση τους σε αυτούς τους πίνακες.

Η εγκατάστασή του γίνεται με την εντολή:

```
pip3 install numpy
```

και η χρήση του με την δήλωση:

```
import numpy as np
```

Το numpy περιέχει ένα μεγάλο πλήθος συναρτήσεων, μερικές από αυτές χρησιμοποιούνται για:

Δημιουργία πινάκων: arange, array, copy, empty, empty_like, eye, fromfile, fromfunction, identity, linspace, logspace, mgrid, ogrid, ones, ones_like, r, zeros, zeros_like

Μετατροπές: ndarray.astype, atleast_1d, atleast_2d, atleast_3d, mat

Χειρισμούς πινάκων: array_split, column_stack, concatenate, diagonal, dsplit, dstack, hsplit, hstack, ndarray.item, newaxis, ravel, repeat, reshape, resize, squeeze, swapaxes, take, transpose, vsplit, vstack

Ερωτήσεις: all, any, nonzero, where

[7] Numpy user guide — SciPy.org, 2019. [Online]. Available: <https://docs.scipy.org/doc/numpy/user/quickstart.html> [Accessed: 7 May 2019]

Ταξινόμηση: argmax, argmin, argsort, max, min, ptp, searchsorted, sort

Πράξεις: choose, compress, cumprod, cumsum, inner, ndarray.fill, imag, prod, put, putmask, real, sum

Στατιστική: cov, mean, std, var

Γραμμική άλγεβρα: cross, dot, outer, linalg.svd, vdot

Στην άσκηση 2.5 όπως είδαμε πιο πάνω γίνεται η εισαγωγή του numpy με το alias np και στην συνέχεια χρησιμοποιούμε την μέθοδο linspace για να δημιουργήσουμε έναν πίνακα 1000 τιμών του S (της διατομής δηλαδή) μεταξύ του 1 και του 30, ώστε με τις τιμές αυτές να υπολογιστεί η αντίστοιχη αντίσταση του αγωγού.

Η linspace λοιπόν επιστρέφει τους αριθμούς που βρίσκονται σε ίσες αποστάσεις σε ένα συγκεκριμένο διάστημα.

```
numpy.linspace(start, stop, num=1000, endpoint=True, retstep=False, dtype=None)
```

Γ. PYCHARM - IDE

Στην ανάπτυξη του Project χρησιμοποιήθηκε ένα από τα πιο γνωστά και διαδεδομένα IDE (Integrated Development Environment) της Python, το PyCharm CE.

Το PyCharm CE (Community Edition) είναι ένα από τα πιο διαδεδομένα IDE (Integrated Development Environment) για τη γλώσσα προγραμματισμού Python. Η JetBrains έχει αναπτύξει το PyCharm ως πλατφόρμα IDE για Python. Εκτός από την υποστήριξη των εκδόσεων 2.x και 3.x της Python, το PyCharm είναι επίσης συμβατό με τα Windows, Linux και macOS. Ταυτόχρονα, τα εργαλεία και τα χαρακτηριστικά γνωρίσματα που παρέχει το PyCharm βοηθούν τους προγραμματιστές να γράψουν μια ποικιλία εφαρμογών λογισμικού στη Python γρήγορα και αποτελεσματικά. Οι προγραμματιστές μπορούν ακόμη και να προσαρμόσουν το PyCharm UI ανάλογα με τις ειδικές ανάγκες και προτιμήσεις τους. Επίσης, μπορούν να επεκτείνουν το IDE επιλέγοντας πάνω από 50 plugins για να ανταποκριθούν στις πολύπλοκες απαιτήσεις του έργου τους.

Η εγκατάσταση του Pycharm CE γίνεται είτε μέσω του repository των Ubuntu, Linux Mint, είτε με τις παρακάτω εντολές μέσα από terminal:

```
sudo apt update
sudo apt install pycharm-community
```

Παρακάτω θα δούμε μερικά βασικά χαρακτηριστικά του:

Code Editor. Ο έξυπνος επεξεργαστής κώδικα που παρέχεται από το PyCharm επιτρέπει στους προγραμματιστές να γράψουν κώδικα Python υψηλής ποιότητας. Ο επεξεργαστής επιτρέπει στους προγραμματιστές να διαβάζουν εύκολα τον κώδικα μέσω των σχημάτων χρωμάτων, να εισάγουν αυτόματα τις εσοχές σε νέες γραμμές, να επιλέγουν το κατάλληλο στυλ κωδικοποίησης και να επωφελούνται από το code completion.

Code Navigation. Οι επιλογές πλοήγησης που παρέχονται από τον PyCharm βοηθούν τους προγραμματιστές να επεξεργάζονται και να βελτιώνουν τον κώδικα χωρίς να βάζουν επιπλέον χρόνο και προσπάθεια. Το IDE διευκολύνει τους προγραμματιστές να

πηγαίνουν σε μια κλάση, ένα αρχείο και ένα σύμβολο, μαζί με τις δηλώσεις που προβάλλονται από μια αναφορά. Ο χρήστης μπορεί να βρει σχεδόν αμέσως ένα στοιχείο στον πηγαίο κώδικα, στο απόσπασμα κώδικα, στο στοιχείο UI. Μπορεί κάποιος επίσης να εντοπίσει περαιτέρω τη χρήση διαφόρων συμβόλων και να ορίσει σελιδοδείκτες στον κώδικα.

Refactoring. Το PyCharm διευκολύνει τους προγραμματιστές να υλοποιούν τόσο τοπικές όσο και καθολικές αλλαγές γρήγορα και αποτελεσματικά. Οι προγραμματιστές μπορούν ακόμη να επωφεληθούν από τις επιλογές refactoring που παρέχονται από το IDE ενώ γράφουν απλό κώδικα Python και δουλεύουν με Python frameworks.

Visual Debugger. Ο Visual Debugger που παρέχεται από το IDE βοηθά τους προγραμματιστές στην αποσφαλμάτωση Python, JavaScript και κώδικα Django. Οι προγραμματιστές μπορούν να χρησιμοποιήσουν το ενσωματωμένο πρόγραμμα εντοπισμού σφαλμάτων για να δουν ζωντανά δεδομένα εντοπισμού σφαλμάτων απευθείας στον επεξεργαστή. Επίσης, μπορούν να εντοπίσουν ταυτόχρονα πολλές διαδικασίες Python και να εξετάσουν τον κώδικα παρακάμπτοντας τις βιβλιοθήκες.

Buid-in Terminal. Το PyCharm συνοδεύεται από τοπικά τερματικά για Windows, Linux και macOS. Το ενσωματωμένο τερματικό επιτρέπει στους προγραμματιστές να συνεχίσουν να γράφουν και να δοκιμάζουν κώδικα χωρίς να εγκαταλείπουν το IDE.

Υποστήριξη Version Control Systems. Το PyCharm επιτρέπει στους προγραμματιστές να δουλεύουν με ευρέως χρησιμοποιούμενα συστήματα ελέγχου εκδόσεων όπως Git, Mercurial, Perforce και SVN. Εκτελεί ακόμη πολύπλοκες εργασίες όπως την αυτόματη προσθήκη, αφαίρεση και διαγραφή αρχείων.[8]

[8] Mindfire Solutions, What is PyCharm IDE?, 2019. [Online]. Available: <https://medium.com/@mindfiresolutions.usa/what-is-pycharm-ide-cc0735784f64> [Accessed: 3 May 2019]

Δ. GIT και GITHUB

Το GitHub είναι η μεγαλύτερη cloud -based πλατφόρμα φιλοξενίας και διαμοιρασμού κώδικα. Είναι μια online υπηρεσία στην οποία οι προγραμματιστές μπορούν να συνδεθούν για να ανεβάσουν ή να κατεβάσουν κώδικα. Το GitHub παρέχει επίσης έλεγχο πρόσβασης και διάφορες λειτουργίες συνεργασίας, όπως wiki και βασικά εργαλεία διαχείρισης εργασιών για κάθε έργο.

Από την άλλη το Git είναι ένα κατανεμημένο ανοιχτού κώδικα σύστημα υπό την άδεια χρήσης GNU GPLv2, που αναπτύχθηκε από τον Linus Torvalds το 2005 με σκοπό την αύξηση της συνεργατικότητας των προγραμματιστών, την δημιουργία αποθετηρίων αλλά και τον έλεγχο των εκδόσεων του κώδικα. Αυτό βασικά σημαίνει ότι το Git παρακολουθεί τις αλλαγές του περιεχομένου ενός αποθετηρίου, καθώς νέος κώδικας προστίθεται από έναν ή περισσότερους προγραμματιστές, διατηρώντας μια ιστορικότητα των αλλαγών αλλά και του χρόνου που αυτές συντελέστηκαν. Το Git δημιουργεί ένα απομακρυσμένο αποθετήριο το οποίο αποθηκεύεται σε έναν Server καθώς και ένα τοπικό αποθετήριο στον υπολογιστή του κάθε προγραμματιστή.

Παρακάτω θα δούμε συνοπτικά την διαδικασία εγκατάστασης του git και την δημιουργία του αποθετηρίου μας στο GitHub.

Δημιουργία λογαριασμού στο GitHub

η σύνδεση μας στο GitHub γίνεται από το παρακάτω url:

<https://github.com/join>

Αφού δημιουργήσαμε και ρυθμίσαμε τον λογαριασμό μας δημιουργούμε το project με όνομα: [Physics_B](#)

Εγκατάσταση του git

Για διανομές λειτουργικών συστημάτων Debian εκτελούμε στο τερματικό τις παρακάτω εντολές:

```
$ sudo apt update
```

```
$ sudo apt install git -y
```

Στην συνέχεια ρυθμίζουμε κάποιες παραμέτρους όπως είναι το username και το email με τα οποία κάναμε register στο GitHub:

```
$ git config --global user.name "username"
```

```
$ git config --global user.email "email@gmail.com"
```

Δημιουργούμε το φάκελο στον οποίο θα αποθηκεύσουμε το αποθετήριο τοπικά στον υπολογιστή μας:

```
$ mkdir workspace
```

και αντιγράφουμε το αποθετήριο από το GitHub:

```
~/workspace/$ git clone https://github.com/geoapos/Physics\_B.git
```

Με την εντολή `git status` εμφανίζεται η κατάσταση του καταλόγου μας (ποια αρχεία είναι ήδη στο Github ποια αρχεία έχουν αλλάξει και ποια είναι εντελώς καινούργια και πρέπει να προστεθούν με την εντολή `git add` . Στην συνέχεια εκτελώντας στο τερματικό `git commit -a` δημιουργούμε την πρώτη έκδοση του project η οποία είναι λειτουργική. Τέλος για να ανεβάσουμε οριστικά στο github τα αρχεία εκτελούμε: `git push origin master`.

Μπορούμε να δούμε της διάφορες εκδόσεις που έχουμε κάνει commit πληκτρολογώντας στον φάκελο του project την εντολή `git log`.

Με βάση την παραπάνω διαδικασία δημιουργήσαμε το αποθετήριο της εργασίας μας το οποίο είναι διαθέσιμο στην διεύθυνση https://github.com/geoapos/Physics_B και για την οποία θα μιλήσουμε αναλυτικά παρακάτω.

E. ΑΠΟΘΕΤΗΡΙΟ

Το αποθετήριο περιλαμβάνει συνολικά 104 λυμένες ασκήσεις σε Python3 των τεσσάρων κεφαλαίων της Φυσικής Β Λυκείου Γενικής Παιδείας.

Το αποθετήριο αποτελείται από:

- Την γραφική διεπαφή χρήστη (GUI), μέσω της οποίας μπορούμε να πλοηγηθούμε στα τέσσερα κεφάλαια ασκήσεων του βιβλίου.
- Τη βιβλιοθήκη των λύσεων των ασκήσεων όπως αυτές παρουσιάζονται στο βιβλίο “Φυσική Γενικής Παιδείας Β · ΤΑΞΗ ΓΕΝΙΚΟΥ ΛΥΚΕΙΟΥ ΛΥΣΕΙΣ ΑΣΚΗΣΕΩΝ”
- Τη βιβλιοθήκη των λύσεων των ασκήσεων σε γλώσσα Python.

E.1. Γραφική διεπαφή χρήστη (GUI)

Τα προγραμματιστικά εργαλεία που χρησιμοποιήθηκαν για την ανάπτυξη της διεπαφής χρήστη είναι η γλώσσα HTML και η γλώσσα CSS. Το βασικό αρχείο της γραφικής διεπαφής είναι το αρχείο index.html.

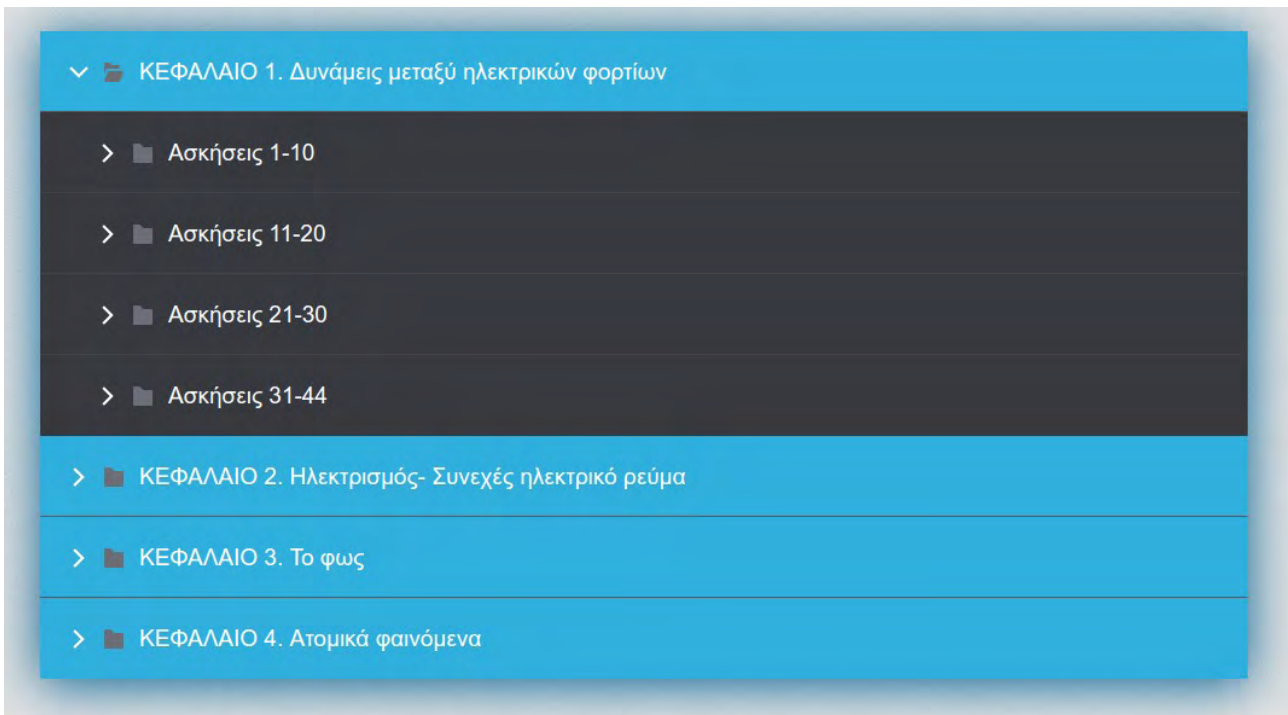
Το menu που δημιουργήσαμε είναι ανεπτυγμένο κατά κεφάλαιο :

κάθε κεφάλαιο είναι χωρισμένο σε ομάδες ασκήσεων, ώστε να γίνεται πιο εύκολη η επιλογή και η προσπέλαση των λύσεων.

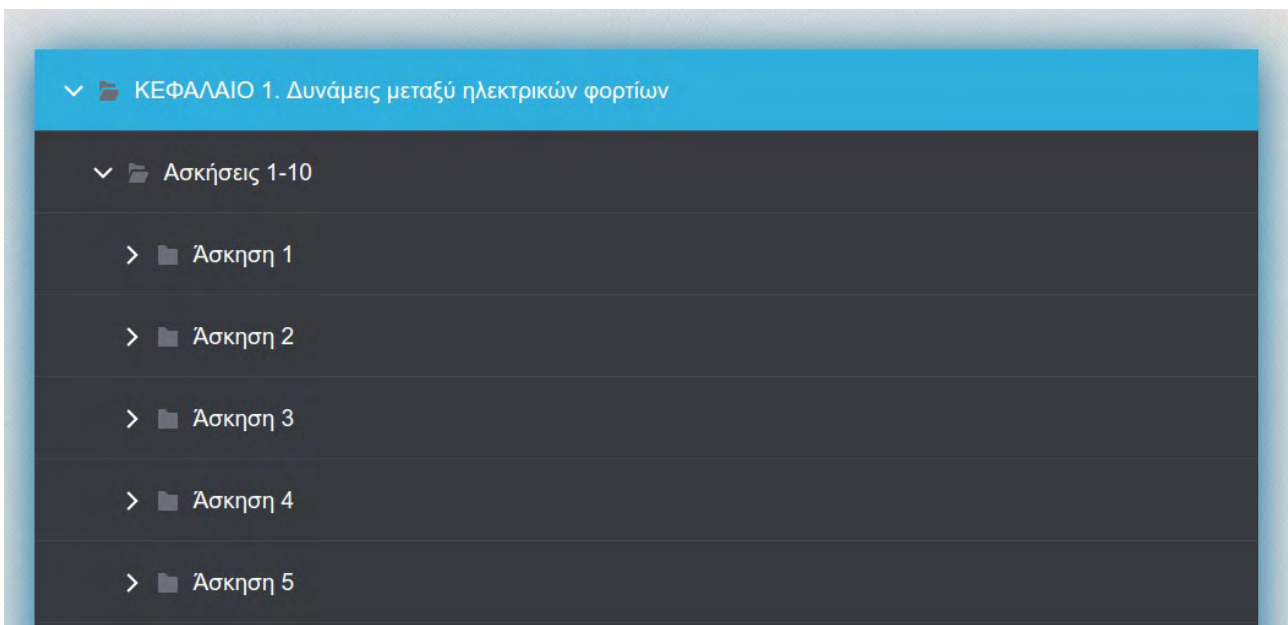
Implementation of programming tools in Cross-thematic approach of Informatics in Physics in Secondary Education.

- > ■ ΚΕΦΑΛΑΙΟ 1. Δυνάμεις μεταξύ ηλεκτρικών φορτίων
- > ■ ΚΕΦΑΛΑΙΟ 2. Ηλεκτρισμός- Συνεχές ηλεκτρικό ρεύμα
- > ■ ΚΕΦΑΛΑΙΟ 3. Το φως
- > ■ ΚΕΦΑΛΑΙΟ 4. Ατομικά φαινόμενα

Η ανάπτυξη του γίνεται με κάθετη αναδίπλωση σε στυλ ακορντεόν:



Από κάθε ομάδα ασκήσεων αναπτύσσεται η επιλογή της κάθε άσκησης:



Και τέλος επιλέγοντας κάθε άσκηση βλέπουμε αριστερά την λύση της άσκησης όπως αυτή παρουσιάζεται στο βιβλίο “Φυσική Γενικής Παιδείας Β΄ ΤΑΞΗ ΓΕΝΙΚΟΥ ΛΥΚΕΙΟΥ ΛΥΣΕΙΣ ΑΣΚΗΣΕΩΝ”[9] και δεξιά των κώδικα της λύσης σε γλώσσα Python3.

The image shows a PDF viewer on the left and a Python code editor on the right. The PDF viewer displays a physics problem (16) about the electric field at the center of a square with four charges at its corners. The problem includes diagrams of the square and the resulting electric field vectors. The Python code on the right implements the solution, calculating the electric field components and magnitude.

Στο αριστερό πεδίο έχουμε την δυνατότητα να κατεβάσουμε σε μορφή pdf την λύση της άσκησης ενώ δεξιά βλέπουμε τον κώδικα Python colorized.

Στην πραγματικότητα για να μπορεί να εμφανίζεται colorized ο κώδικας έχουν γίνει εισαγωγή object αρχεία html στο κύριο αρχείο index.html. Αυτά τα αρχεία html κάθε άσκησης έχουν παραχθεί από τα αρχεία του κώδικα κάθε άσκησης με την χρήση ενός άλλου πακέτου της Python, το **Pygments**.

Το pygments είναι ένα πακέτο της Python που μας παρέχει την λειτουργικότητα για μια χρωματική ανάδειξη της σύνταξης κώδικα, κατάλληλο για χρήση σε εφαρμογές φιλοξενίας κώδικα όπως forums, wiki ή άλλες εφαρμογές που χρειάζονται να εμπλουτιστούν με πηγαίο κώδικα.

[9] Φυσική Γενικής Παιδείας Β΄ Τάξη Γενικού Λυκείου, Λύσεις Ασκήσεων, ΙΝΣΤΙΤΟΥΤΟ ΤΕΧΝΟΛΟΓΙΑΣ ΥΠΟΛΟΓΙΣΤΩΝ ΚΑΙ ΕΚΔΟΣΕΩΝ «ΔΙΟΦΑΝΤΟΣ», Αθήνα 2013

Η εγκατάσταση του σε ένα Debian-like σύστημα γίνεται με την εντολή:

```
pip3 install pygments
```

Η μετατροπή των αρχείων κώδικα Python (py) έγινε αυτοματοποιημένα με την χρήση του script pygm.py:

```
#!/usr/bin/python3
#filename pygm.py
"""
Script με το οποίο δημιουργούνται τα html files μέσω της pygmentize
python3 pygm.py [path]
"""

import os
import sys

if len(sys.argv)==2:
    folder=sys.argv[1]

    if os.path.exists(folder):          #check if folder exist
        os.chdir(folder)              #change directory where the files are
        files=os.listdir()            #list of all files in current directory

        for file in files:
            if (file[-3:]) == '.py':   #Αν το αρχείο έχει extention .py
                if os.path.isfile((file[:-3]+'py.html')): #Αν υπάρχει ήδη μέσα στο path συνέχισε στο επόμενο
                    continue
                else:                   #Αν δεν υπάρχει ήδη μέσα στο path
                    filetemp=file.strip('.py') #strip py from filename
                    os.system('pygmentize -O full,style=emacs -o {}py.html {}'.format(filetemp, file)) #εκτέλεσε την
                    εντολή

            else: print('Sorry, directory does not exist, exiting...')

        else:
            print('You must specify the path')
```

και με την εντολή σε τερματικό:

```
python3 pygm.py [path]
```

όπου [path] το path που βρίσκονται τα αρχεία py των ασκήσεων.

E.2. HTML

Όπως αναφέρθηκε και προηγουμένως το GUI του αποθετηρίου βασίζεται στο αρχείο index.html και δημιουργήθηκε φυσικά με την χρήση της HTML5.

Στο κυρίως σώμα (body) του αρχείου δημιουργούμε μια Unordered HTML List με την χρήση των ετικετών : και για να παρουσιάσουμε τα κεφάλαια του μαθήματος και τις ασκήσεις του κάθε κεφαλαίου, έτσι π.χ.:

με την επαναλαμβανόμενη χρήση του παρακάτω κώδικα δημιουργούμε τα 4 κεφάλαια

```
<ul class="cd-accordion-menu animated">
  <!-- ΚΕΦΑΛΑΙΟ 1 -->
  <li class="has-children">
    <input type="checkbox" name ="chapter1" id="chapter1">
    <label for="chapter1">ΚΕΦΑΛΑΙΟ 1. Δυνάμεις μεταξύ ηλεκτρικών φορτίων</label>
```

όπως επίσης και με την χρήση του παρακάτω κώδικα δημιουργούμε την λίστα των ασκήσεων.

```
<ul>
  <li class="has-children">
    <!-- Άσκηση 1 -->
    <input type="checkbox" name ="chapter1-1" id="chapter1-1">
    <label for="chapter1-1">Άσκηση 1 </label>
    <ul>
      <embed src="Docs/pdf/Chapter1/1_1.pdf#view=FitH" width="45%" height=500px
type="application/pdf" style="display: inline-block;">
      <object data="Docs/py/Chapter1/1_1py.html" width="55%" height=500px type="text/html"
border="0" style="float: right;" </object>
    </ul>
  </li>
```

το pdf αρχείο με την λύση της άσκησης σύμφωνα με το βιβλίο λύσεων του Υπουργείου έχει ενσωματωθεί με την χρήση της ετικέτας <embed> η οποία ενσωματώνει ένα εξωτερικό περιεχόμενο στο καθορισμένο σημείο του εγγράφου. Αυτό το περιεχόμενο μπορεί να παρέχεται από εξωτερική εφαρμογή ή άλλη πηγή διαδραστικού περιεχομένου.

Ο κώδικας της Python που αντιπαραβάλλεται δεξιά ενσωματώνεται με την χρήση της ετικέτας <object>. Η ετικέτα <object> είναι μια ετικέτα HTML και χρησιμοποιείται για την

προβολή πολυμέσων όπως ηχητικά μηνύματα, βίντεο, εικόνες, αρχεία PDF και Flash σε ιστοσελίδες. Μπορεί επίσης να χρησιμοποιηθεί για την προβολή μιας άλλης ιστοσελίδας (δηλαδή ενός άλλου html αρχείου) μέσα στη σελίδα HTML όπως στην δική μας περίπτωση που ενσωματώσαμε τα αρχεία html των λύσεων σε ργthon που δημιουργήσαμε με την pygmentize όπως είδαμε παραπάνω.

ΣΤ. ΦΥΣΙΚΗ ΓΕΝΙΚΗΣ ΠΑΙΔΕΙΑΣ Β ΛΥΚΕΙΟΥ

Η Φυσική γενικής Παιδείας της Β' Λυκείου αποτελείται από 4 κεφάλαια. Στόχος του μαθήματος είναι οι μαθητές να [10].:

- γνωρίσουν την επιστημονική μέθοδο
- προσεγγίσουν ποιοτικά, ποσοτικά και πειραματικά βασικές έννοιες και νόμους της Φυσικής.
- Γνωρίσουν τη χρήση των ποσοτικών μεθόδων και του πειραματισμού και να αναπτύξουν πρακτικές δεξιότητες με το χειρισμό οργάνων, διατάξεων και συσκευών.
- Αποσαφηνίσουν τις επιστημονικές τους ιδέες, αξιοποιώντας την επιστημονική ορολογία εκφραζόμενοι στην καθημερινή γλώσσα.
- Καλλιεργήσουν νοητικές δεξιότητες για την αντιμετώπιση προβλημάτων, αναπτύσσοντας κριτική σκέψη, δημιουργική φαντασία και ικανότητα επικοινωνίας και να ανακαλούν στην μνήμη τους ένα πρόβλημα του οποίου γνωρίζουν τη λύση και που το θεωρούν ανάλογο με το προς επίλυση πρόβλημα.
- Αναγνωρίσουν τον ουσιαστικό ρόλο που παίζει η Φυσική επιστήμη σε όλο το φάσμα της εμπειρίας τους (δεξιότητες και στάσεις) και των γνώσεών τους, από την καθημερινή ζωή ως τις βασικές λειτουργίες του σύμπαντος.

ΣΤ.1. Κεφάλαιο 1. ΔΥΝΑΜΕΙΣ ΜΕΤΑΞΥ ΗΛΕΚΤΡΙΚΩΝ ΦΟΡΤΙΩΝ

Περιεχόμενο:

- Ο νόμος του Coulomb

[10] Πρόγραμμα Σπουδών του μαθήματος «Φυσική» της Α' και Β' τάξης Γενικού Λυκείου και της ομάδας προσανατολισμού Θετικών Σπουδών της Β' και Γ' τάξης Γενικού Λυκείου., ΦΕΚ Β 184/23-01-2015.

- Ηλεκτρικό πεδίο - Ένταση - Δυναμικές γραμμές
- Ηλεκτρική δυναμική ενέργεια
- Δυναμικό - Διαφορά δυναμικού
- Πυκνωτές

Κύριοι μαθησιακοί στόχοι:

Οι μαθητές να:

- συνδέουν καθημερινά φαινόμενα στατικού ηλεκτρισμού με την έννοια του ηλεκτρικού φορτίου
- αναγνωρίζουν την ύπαρξη δύο ειδών φορτίων
- συνδέουν το ηλεκτρικό φορτίο με τη δομή του ατόμου
- διακρίνουν την ύλη ανάλογα με την αγωγιμότητα που παρουσιάζει σε αγωγούς, μονωτές και ημιαγωγούς
- περιγράφουν ποσοτικά την ηλεκτρική αλληλεπίδραση με την δύναμη Coulomb
- σχεδιάζουν τις δυνάμεις που ασκούνται σε σύστημα δύο φορτίων
- αναγνωρίζουν τα χαρακτηριστικά της δύναμης Coulomb
- περιγράφουν το χώρο γύρω από το φορτίο αξιοποιώντας την έννοια του πεδίου
- αντιληφθούν την ένταση του ηλεκτρικού πεδίου σε ένα σημείο ως συνάρτηση της θέσης του σημείου.
- διακρίνουν τις διάφορες μορφές ηλεκτροστατικών πεδίων ανάλογα της μεταβολής ή όχι της έντασης του ηλεκτρικού πεδίου.
- Αναγνωρίζουν και να αξιοποιούν την δυναμική ενέργεια ως ενέργεια κατάστασης μεταξύ φορτισμένων σωματιδίων.

- Συνδέουν την διαφορά δυναμικού με την μεταβολή της δυναμικής ενέργειας.

ΣΤ.2. Κεφάλαιο 2. ΗΛΕΚΤΡΙΣΜΟΣ: ΣΥΝΕΧΕΣ ΗΛΕΚΤΡΙΚΟ ΡΕΥΜΑ

Περιεχόμενο:

- Ηλεκτρικές πηγές
- Ηλεκτρικό ρεύμα
- Κανόνες του Kirchhoff
- Αντίσταση (ωμική) - Αντιστάτης
- Συνδεσμολογία αντιστατών (αντιστάσεων)
- Ρυθμιστική (μεταβλητή) αντίσταση
- Ενέργεια και ισχύς του ηλεκτρικού ρεύματος
- Ηλεκτρεγερτική δύναμη (ΗΕΔ) πηγής
- Νόμος του Ohm για κλειστό κύκλωμα
- Αποδέκτες
- Δίοδος

Κύριοι μαθησιακοί στόχοι:

Οι μαθητές να:

- να συνθέτουν ένα απλό ηλεκτρικό κύκλωμα
- να αναγνωρίζουν και να κατανοήσουν την λειτουργία των οργάνων μέτρησης (βολτόμετρα και αμπερόμετρα).
- Να συνδέουν την τιμή μιας αντίστασης με τα φυσικά και γεωμετρικά χαρακτηριστικά του αγωγού.
- να αξιοποιούν τον νόμο του Ohm στην επίλυση ηλεκτρικών κυκλωμάτων.

- Να υπολογίζουν σύνθετες αντιστάσεις
- Να χρησιμοποιούν τους κανόνες Kirchhoff στην επίλυση των ηλεκτρικών προβλημάτων
- εκτελούν υπολογισμούς ενέργειας και ισχύος σε ένα ηλεκτρικό κύκλωμα.

ΣΤ.3. Κεφάλαιο 3. ΦΩΣ

Περιεχόμενο:

- Η φύση του φωτός
- Η ταχύτητα του φωτός
- Μήκος κύματος και συχνότητα φωτός κατά τη διάδοσή του
- Ανάλυση λευκού φωτός και χρώματα
- Πόλωση του φωτός

Κύριοι μαθησιακοί στόχοι:

Οι μαθητές να:

- κατανοήσουν την δυαδική μορφή των φωτονίων, δηλαδή τη φύσης του ως κύμα αλλά και ως σωματίδιο.
- Αντιληφθούν την ταχύτητα του φωτός ως τιμή και ως όριο τιμής ταχυτήτων
- σχεδιάζουν την πορεία φωτεινών ακτίνων σε ένα ομογενές μέσο
- διακρίνουν τα είδη ηλεκτρομαγνητικής ακτινοβολίας ανάλογα με την συχνότητα της.
- Αναπαραστούν το φάσμα της ορατής ακτινοβολίας και να αναγνωρίζουν τις χαρακτηριστικές περιοχές του Η/Μ φάσματος
- κατανοήσουν ότι ενέργεια που μεταφέρει το φως είναι ενέργεια ηλεκτρικού και μαγνητικού Πεδίου.
- γνωρίζουν τι είναι τα πολωτικά φίλτρα ή οι πολωτές.

ΣΤ.4. Κεφάλαιο 4. ΑΤΟΜΙΚΑ ΦΑΙΝΟΜΕΝΑ

Περιεχόμενο:

- Ενέργεια του ηλεκτρονίου στο άτομο του υδρογόνου
- Διακριτές ενεργειακές στάθμες
- Μηχανισμός παραγωγής και απορρόφησης φωτονίων
- Ακτίνες Χ

Κύριοι μαθησιακοί στόχοι:

Οι μαθητές να:

- αναγνωρίζουν το ηλεκτρικό φορτίο του ηλεκτρονίου ως στοιχειώδη ποσότητα φορτίου.
- Γνωρίζουν την εξέλιξη των ιδεών πάνω σε ατομικά μοντέλα
- περιγράφουν από τι συνίσταται ο πυρήνας
- αναφέρουν τα χαρακτηριστικά των σωματιδίων του πυρήνα
- κατανοήσουν τις έννοιες της διέγερσης και του ιονισμού του ατόμου.
- γνωρίζουν για τις ακτίνες Χ, τον τρόπο παραγωγής τους τις χαρακτηριστικές τους ιδιότητες.

Z. ΑΝΑΦΟΡΕΣ- ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] Heidi Hayes Jacobs, Interdisciplinary Curriculum: Design and Implementation, ISBN-0-87120-165-8, 1989
- [2] Μασταγγούρας Ηλίας, Η διαθεματικότητα στη σχολική γνώση, ISBN 960-333-310-7, 2002
- [3] Python Documentation by Version, 2019. [Online]. Available: <https://www.python.org/doc/versions/>
[Accessed: 2 May 2019]
- [4] Eric Matthes, "Python Crash Course, A Hands-On, Project-Based Introduction to Programming", San Francisco: William Pollock, 2016
- [5] Δρ. Νικόλαος Ζάχαρης, Δρ. Γεώργιος Μαυρομμάτης, "Εκπαίδευση στην Python", Αθήνα: Εθνικό Κέντρο Δημόσιας Διοίκησης & Αυτοδιοίκησης, 2014
- [6] math — Mathematical functions, Python 3.7.3 documentation, 2019. [Online]. Available: <https://docs.python.org/3/library/math.html> [Accessed: 4 May 2019]
- [7] Numpy user guide — SciPy.org, 2019. [Online]. Available: <https://docs.scipy.org/doc/numpy/user/quickstart.html> [Accessed: 7 May 2019]
- [8] Mindfire Solutions, What is PyCharm IDE?, 2019. [Online]. Available: <https://medium.com/@mindfiresolutions.usa/what-is-pycharm-ide-cc0735784f64> [Accessed: 3 May 2019]
- [9] Φυσική Γενικής Παιδείας Β΄ Τάξη Γενικού Λυκείου, Λύσεις Ασκήσεων, ΙΝΣΤΙΤΟΥΤΟ ΤΕΧΝΟΛΟΓΙΑΣ ΥΠΟΛΟΓΙΣΤΩΝ ΚΑΙ ΕΚΔΟΣΕΩΝ «ΔΙΟΦΑΝΤΟΣ», Αθήνα 2013
- [10] Πρόγραμμα Σπουδών του μαθήματος «Φυσική» της Α΄ και Β΄ τάξης Γενικού Λυκείου και της ομάδας προσανατολισμού Θετικών Σπουδών της Β΄ και Γ΄ τάξης Γενικού Λυκείου., ΦΕΚ Β 184/23-01-2015.