

# 1η Υποχρεωτική Εργασία LaTeX

Ονοματεπώνυμο: Στέφανος Καραμπέρας  
AEM: 2910

16 Δεκεμβρίου 2019

## Άσκηση 1

### Γενικές παρατηρήσεις

Το προγραμματιστικό μέρος της εργασίας έχει υλοποιηθεί με χρήση Πιπτην 3. Ο κώδικας που έχει συνταχθεί για την άσκηση 1 βρίσκεται εντός του φακέλου “Code/task-1” .

Οι ζητούμενοι αλγόριθμοι που εφαρμόζουν τις τεχνικές προσέγγισης ρίζας με τη μέθοδο της διχοτόμησης, τη μέθοδο Newton-Raphson και τη μέθοδο της τέμνουσας, έχουν υλοποιηθεί εντός του αρχείου “root\_estimation\_algorithms.py”. Επιπρόσθετα, εντός του αρχείου f\_function.py έχουν δημιουργηθεί οι συναρτήσεις calculate\_f(x), calculate\_der\_1\_f(x) και calculate\_der\_2\_f(x). Οι συναρτήσεις αυτές δέχονται ως όρισμα έναν αριθμό  $\xi$  και επιστρέφουν την ρίζα της  $f(x)$ ,  $f'(x)$  και  $f''(x)$  αντίστοιχα, για την δοθείσα συνάρτηση  $f(x)$  της εκφώνησης της άσκησης 1. Τέλος, με δεδομένο πως η ζητούμενη ακρίβεια ορίζεται στο 5ο δεκαδικό ψηφίο, η μεταβλητή t\_err στο αρχείο “root\_estimation\_algorithms.py” αρχικοποιείται με την τιμή  $t\_err = 0.000005$  (ακρίβεια 5ου δεκαδικού ψηφίου με στρογγυλοποίηση).

### Αλγόριθμος για τη μέθοδο Διχοτόμησης

Η συνάρτηση που υλοποιήθηκε για την εκτέλεση του αλγορίθμου που εφαρμόζει τη μέθοδο της διχοτόμησης είναι η “partitioning\_root\_estimation()”.

Η συνάρτηση αρχικά υπολογίζει το πλήθος των απαιτούμενων επαναλήψεων της μεθόδου διχοτόμησης για την επίτευξη της ζητούμενης ακρίβειας. Στη συνέχεια, ξεκινά να εκτελεί το ορισμένο πλήθος επαναλήψεων, σταματώντας σε περίπτωση που βρει ακριβή ρίζα της  $f(x)$  και τυπώνοντας τη ρίζα, ή συνεχίζοντας μέχρι να ολοκληρώσει το ορισμένο πλήθος επαναλήψεων, τυπώνοντας τελικά την προσέγγιση της ρίζας και το πλήθος επαναλήψεων που απαιτήθηκαν για την εκτίμησή της.

## Αλγόριθμος για τη μέθοδο Newton-Raphson

Η συνάρτηση που υλοποιήθηκε για την εκτέλεση του αλγορίθμου που εφαρμόζει τη μέθοδο Newton-Raphson είναι η `newton_raphson_root_estimation()`.

Η συνάρτηση αρχικοποιεί το  $x_{n-1}$  χρησιμοποιώντας το πρώτο στοιχείο του πεδίου ορισμού  $[-2,2]$  της  $f(x)$ , δηλαδή το  $-2$ . Ακολουθώντας, υπολογίζει διαδοχικά το  $x_n$  μέχρις ότου η απόλυτη τιμή της διαφοράς  $x_{n-1}-x_{n-1}$  να είναι μικρότερη ή ίση του καθορισμένου αποδεκτού σφάλματος (ακρίβεια), δηλαδή  $|x_n - x_{n-1}| \leq t\_err$ . Στο τέλος κάθε επανάληψης, η τιμή του  $q_{n-1}$  ανανεώνεται με την τιμή του  $q_n$ . Η συνάρτηση καταγράφει σε μια μεταβλητή `iter_count` το πλήθος επαναλήψεων που πραγματοποιεί κατά την εκτέλεσή της. Με τη λήξη των επαναλήψεων λόγω επίτευξης της επιθυμητής ακρίβειας, τυπώνεται το πλήθος επαναλήψεων που εκτελέστηκαν και η προσέγγιση της ρίζας που υπολογίστηκε.

## Αλγόριθμος για τη μέθοδο Τέμνουσας

Η συνάρτηση που υλοποιήθηκε για την εκτέλεση του αλγορίθμου που εφαρμόζει τη μέθοδο της τέμνουσας είναι η `secant_root_estimation()`.

Η συνάρτηση αρχικοποιεί το  $x_{n-1}$  χρησιμοποιώντας το πρώτο στοιχείο του πεδίου ορισμού  $[-2,2]$  της  $f(x)$ , δηλαδή το  $-2$ . Επιπρόσθετα, αρχικοποιεί το  $x_n$  χρησιμοποιώντας το τελευταίο στοιχείο του πεδίου ορισμού  $[-2,2]$  της  $f(x)$ , δηλαδή το  $2$ .

Ακολουθώντας, υπολογίζει διαδοχικά το  $x_{n+1}$  μέχρις ότου η απόλυτη τιμή της διαφοράς  $x_{n+1} - x_n$  να είναι μικρότερη ή ίση του καθορισμένου αποδεκτού σφάλματος (ακρίβεια), δηλαδή  $|x_{n+1} - x_n| \leq t\_err$ . Στο τέλος κάθε επανάληψης, η τιμή του  $q_{n-1}$  ανανεώνεται με την τιμή του  $q_n$ , ενώ η τιμή του  $q_n$  ανανεώνεται με την τιμή του  $q_{n+1}$ . Η συνάρτηση καταγράφει σε μία μεταβλητή `iter_count` το πλήθος επαναλήψεων που πραγματοποιεί κατά την εκτέλεσή της. Με τη λήξη των επαναλήψεων λόγω επίτευξης της επιθυμητής ακρίβειας, τυπώνεται το πλήθος επαναλήψεων που εκτελέστηκαν και η προσέγγιση της ρίζας που υπολογίστηκε.

## Εκτέλεση των αλγορίθμων και αποτελέσματα

Από την εκτέλεση του αλγορίθμου για τη μέθοδο διχοτόμησης, προκύπτει ως λύση η ακριβής ρίζα  $0.0$  της  $f(x)$ .

Από την εκτέλεση του αλγορίθμου για τη μέθοδο Newton-Raphson, προκύπτει ως λύση η προσεγγιστική ρίζα  $-1.19762372213359$  με την εκτέλεση  $7$  επαναλήψεων.

Από την εκτέλεση του αλγορίθμου για τη μέθοδο της τέμνουσας, προκύπτει ως λύση η προσεγγιστική ρίζα  $1.5301335105281786$  με την εκτέλεση  $10$  επαναλήψεων.

## Άσκηση 2

### Γενικές παρατηρήσεις

Ο κώδικας που έχει συνταχθεί για την άσκηση 2 βρίσκεται εντός του φακέλου “Code/task-2” .

Οι ζητούμενοι αλγόριθμοι που εφαρμόζουν τις τεχνικές προσέγγισης ρίζας με την τροποποιημένη μέθοδο Newton-Raphson, την τροποποιημένη μέθοδο διχοτόμησης και την τροποποιημένη μέθοδο της τέμνουσας, έχουν υλοποιηθεί εντός του αρχείου “modified\_root\_estimation\_algorithms.py” Επιπρόσθετα, εντός του αρχείου f\_function.py έχουν δημιουργηθεί οι συναρτήσεις calculate\_f(x), calculate\_der\_1\_f(x) και calculate\_der\_2\_f(x). Οι συναρτήσεις αυτές δέχονται ως όρισμα έναν αριθμό  $\xi$  και επιστρέφουν την ρίζα της  $f(x)$ ,  $f'(x)$  και  $f''(x)$  αντίστοιχα, για την δοθείσα συνάρτηση  $f(x)$  της εκφώνησης της άσκησης 2. Τέλος, με δεδομένο πως η ζητούμενη ακρίβεια ορίζεται στο 5ο δεκαδικό ψηφίο, η μεταβλητή t\_err στο αρχείο “root\_estimation\_algorithms.py” αρχικοποιείται με την τιμή t\_err = 0.000005 (ακρίβεια 5ου δεκαδικού ψηφίου με στρογγυλοποίηση).

### Αλγόριθμος για την τροποποιημένη μέθοδο Newton-Raphson

Η συνάρτηση που υλοποιήθηκε για την εκτέλεση του αλγορίθμου που εφαρμόζει τη μέθοδο Newton-Raphson είναι η `modified_newton_raphson_root_estimation(limit_of_iterations=No`

Η συνάρτηση αρχικοποιεί το  $x_n$  χρησιμοποιώντας το πρώτο στοιχείο του πεδίου ορισμού  $[-2,2]$  της  $f(x)$ , δηλαδή το -2. Ακολουθώντας, υπολογίζει διαδοχικά το  $x_{n+1}$  μέχρις ότου η απόλυτη τιμή της διαφοράς  $x_{n+1}-x_n$  να είναι μικρότερη ή ίση του καθορισμένου αποδεκτού σφάλματος (ακρίβεια), δηλαδή  $|x_{n+1} - x_n| \leq t\_err$ . Η συνάρτηση καταγράφει σε μια μεταβλητή iter\_count το πλήθος επαναλήψεων που εκτελεί κατά την εκτέλεσή της. Με τη λήξη των επαναλήψεων λόγω επίτευξης της επιθυμητής ακρίβειας, τυπώνεται το πλήθος επαναλήψεων που εκτελέστηκαν και η προσέγγιση της ρίζας που υπολογίστηκε.

### Αλγόριθμος για την τροποποιημένη μέθοδο διχοτόμησης

Η συνάρτηση που υλοποιήθηκε για την εκτέλεση του αλγορίθμου που εφαρμόζει την τροποποιημένη μέθοδο της διχοτόμησης είναι η `modified_partitioning_root_estimation()`. Αρχικά, η συνάρτηση υπολογίζει το πλήθος των απαιτούμενων επαναλήψεων της μεθόδου διχοτόμησης για την επίτευξη της ζητούμενης ακρίβειας. Στη συνέχεια, ξεκινά να εκτελεί το ορισμένο πλήθος επαναλήψεων, σταματώντας σε περίπτωση που βρει ακριβή ρίζα της  $f(x)$  και τυπώνοντας τη ρίζα, ή συνεχίζοντας μέχρι να ολοκληρώσει το ορισμένο πλήθος επαναλήψεων, τυπώνοντας τελικά την προσέγγιση της ρίζας και το πλήθος επαναλήψεων που απαιτήθηκαν για την εκτίμησή της.

Κατά την εκτέλεση των επαναλήψεων, η επιλογή του τυχαίου σημείου m που χρησιμοποιείται στον υπολογισμό της προσεγγιστικής τιμής  $f(m)$  γίνεται με

χρήση της γεννήτριας τυχαίων αριθμών της βιβλιοθήκης ρανδομ που είναι προεγκατεστημένη στην Πψτην. Συγκεκριμένα, η εντολή `random.uniform(Ffunction.calc_space[0], Ffunction.calc_space[1])` επιλέγει σε κάθε της εκτέλεση έναν τυχαίο δεκαδικό αριθμό στο διάστημα  $[-2,2]$ .

### Αλγόριθμος για την τροποποιημένη μέθοδο τέμνουσας

Η συνάρτηση που υλοποιήθηκε για την εκτέλεση του αλγορίθμου που εφαρμόζει την τροποποιημένη μέθοδο της τέμνουσας είναι η `“modified_secant_root_estimation()”`.

Η συνάρτηση αρχικοποιεί το  $x_n$  χρησιμοποιώντας το πρώτο στοιχείο του πεδίου ορισμού  $[-2,2]$  της  $f(x)$ , δηλαδή το  $-2$ . Επιπρόσθετα, αρχικοποιεί το  $x_{n+1}$  χρησιμοποιώντας το σημείο  $x = 0.1$ , και το  $x_{n+2}$  χρησιμοποιώντας το τελευταίο στοιχείο του πεδίου ορισμού  $[-2,2]$  της  $f(x)$ , δηλαδή το  $2$ .

Ακολουθώντας, υπολογίζει διαδοχικά το  $x_{n+3}$  μέχρις ότου η απόλυτη τιμή της διαφοράς  $x_{n+3} - x_{n+2}$  να είναι μικρότερη ή ίση του καθορισμένου αποδεκτού σφάλματος (ακρίβεια), δηλαδή  $|x_{n+3} - x_{n+2}| \leq t\_err$ . Η συνάρτηση καταγράφει σε μια μεταβλητή `υπαδε_ξ_ινδ` ποιό από τα  $x_n, x_{n+1}, x_{n+2}$  πρέπει να ανανεωθεί κάθε φορά παίρνοντας την τιμή του  $x_{n+3}$ , ανάλογα με το ποιό από τα 3 σημεία προαναφερθέντα σημεία είναι παλιότερο. Επίσης, μια μεταβλητή `iter_count` αξιοποιείται για την καταγραφή του πλήθους επαναλήψεων που πραγματοποιούνται κατά την εκτέλεση της μεθόδου.

Με τη λήξη των επαναλήψεων λόγω επίτευξης της επιθυμητής ακρίβειας, τυπώνεται το πλήθος επαναλήψεων που εκτελέστηκαν και η προσέγγιση της ρίζας που υπολογίστηκε.

### Εκτέλεση των αλγορίθμων και αποτελέσματα

#### Ερώτημα 1.

Από την εκτέλεση του αλγορίθμου για την τροποποιημένη μέθοδο Newton-Raphson, προκύπτει ως λύση η προσεγγιστική ρίζα  $-1.3812984883894897$  με την εκτέλεση 4 επαναλήψεων.

Από την εκτέλεση του αλγορίθμου για την τροποποιημένη μέθοδο διχοτόμησης, προκύπτει ως λύση η προσεγγιστική ρίζα  $-1.3811460463149443$  με την εκτέλεση 19 επαναλήψεων. Σημειώνεται σε αυτό το σημείο πως λόγω της χρήσης γεννήτριας τυχαίων αριθμών, το αποτέλεσμα αυτό συχνά υφίσταται τυχαίες διακυμάνσεις σε κάθε εκτέλεση του αλγορίθμου, όσον αφορά την τιμή της προσεγγιστικής ρίζας.

Από την εκτέλεση του αλγορίθμου για την τροποποιημένη μέθοδο της τέμνουσας, προκύπτει ως λύση η προσεγγιστική ρίζα  $0.20518292468887597$  με την εκτέλεση 6 επαναλήψεων.

## Ερώτημα 2.

Εκτελώντας 10 επαναλήψεις του τροποποιημένου αλγορίθμου διχοτόμησης, λαμβάνουμε τα παρακάτω αποτελέσματα:

1η επανάληψη: Finished after 19 approximation iterations. Approximation is: -1.381790712760268

2η επανάληψη: Finished after 19 approximation iterations. Approximation is: -1.381475154120745

3η επανάληψη: Finished after 19 approximation iterations. Approximation is: 1.9999500173012472

4η επανάληψη: Finished after 19 approximation iterations. Approximation is: 1.999999979070466

5η επανάληψη: Finished after 19 approximation iterations. Approximation is: -1.3830710263393167

6η επανάληψη: Finished after 19 approximation iterations. Approximation is: -1.3813186273722509

7η επανάληψη: Finished after 19 approximation iterations. Approximation is: -1.3799733240691971

8η επανάληψη: Finished after 19 approximation iterations. Approximation is: -1.3814260841319628

9η επανάληψη: Finished after 19 approximation iterations. Approximation is: -1.3817409842964017

10η επανάληψη: Finished after 19 approximation iterations. Approximation is: 1.9999999593232018

Παρατηρούμε ότι και στις 10 εκτελέσεις του αλγορίθμου, η σύγκλιση στην προσεγγιστική ρίζα προκύπτει μέσα από 19 επαναλήψεις. Συνεπώς, ο αριθμός επαναλήψεων παραμένει σταθερός.

## 1 Άσκηση 1 (Λύση)

$_{AB\Gamma\Delta EZH\Theta I\kappa\lambda\mu\nu\xi\omicron\pi\rho\varsigma}$

## 2 Άσκηση 2 (Λύση)

*Normal Italics* **Bold**  
*Emphasized* Underlined

## 3 Άσκηση 3 (Λύση)

$$a^2 + b^2 = c^2$$

$$e^{i\pi} = -1$$

$$\pi = \frac{c}{d}$$

$$\frac{d}{dx} \int_a^x f(s) ds = f(x)$$

$$f(x) = \sum_{i=0}^{\infty} \frac{f^{(i)}(0)}{i!} x^i$$

$$\mathbf{Ax} = \mathbf{b}$$

$$\|x + y\| \leq \|x\| + \|y\|$$

$$\mathbf{I} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (1)$$

$$\mathbf{I} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

$$\mathbf{I} = \left\{ \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \right\}, \quad \mathbf{I} = \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}, \quad \mathbf{I} = \left\| \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \right\| \quad (3)$$

*Τέφας*      2   3  
*Πήτας*      5   6  
*Λάσκαρης*   8   9

<i>Κοτρόπουλος</i>	2	3
<i>Πήτας</i>	5	6
<i>Νικολαΐδης</i>	8	9

1	2	3
4	5	6
7	8	9

1	2	3
4	5	6
7	8	9

#### 4 Άσκηση 4 (Λύση)

<i>Μέλη ΔΕΠ Πληροφορικής</i>		
<i>Λέκτορες</i>	<i>VD</i>	<i>Δραζιώτης Κωνσταντίνος</i>
<i>Επικουροι</i>	<i>LN</i>	<i>Λάσκαρης Νικόλαος</i>
	<i>TG</i>	<i>Τσουμάκας Γρηγόριος</i>
<i>Αναπληρωτές</i>	<i>TA</i>	<i>Τέφας Αναστάσιος</i>
	<i>PN</i>	<i>Πλέρος Νίκος</i>
	<i>PA</i>	<i>Παπαδόπουλος Απόστολος</i>
<i>Καθηγητές</i>	<i>KC</i>	<i>Κοτρόπουλος Κωνσταντίνος</i>
	<i>PI</i>	<i>Πήτας Ιωάννης</i>
	<i>VI</i>	<i>Βλαχάβας Ιωάννης</i>

#### 5 Άσκηση 5 (Λύση)

- *Τέφας*
- *Μπουζάς*
- *Μπρούζα*
- *Λάσκαρης*
- *Κοτρόπουλος*
- *Πήτας*
- *Νικολαΐδης*

1. Τέφας
2. Μπουζάς
3. Μπρούζα
4. Λάσκαρης
5. Κοτρόπουλος
6. Πήτας
7. Νικολαΐδης

- (α) Τέφας
- (β) Μπουζάς
- (γ) Μπρούζα
- (δ) Λάσκαρης
- (ε) Κοτρόπουλος
- (ζ) Πήτας
- (η) Νικολαΐδης