Η Ομάδα μας 57

Κωνσταντινος Χουλιαρας, 1505

Ιωαννης Ασται, 1448

Εργασία στις Ευκλείδειες και τροποποιημένες Ευκλείδειες πολυωνυμικές ακολουθίες (prs)

1 Εισαγωγή

Τα πολυώνυμα στις Ευκλείδειες και τροποποιημένες Ευκλείδειες ακολουθίες είναι τα υπόλοιπα από Ευκλείδειες ή τροποποιημένες Ευκλείδειες διαιρέσεις πολυωνύμων (όπου αλλάζουμε το πρόσημο των υπολοίπων). Τα πολυώνυμα αυτά προκύπτουν κατά τον υπολογισμό του μκδ ή τροποποιημένου μκδ πολυωνύμων $f,\ g \in Z[x]$ ή $f,\ g \in Q[x]$. Ο Ευκλείδειος ή τροποποιημένος Ευκλείδειος αλγόριθμος πολυωνύμων μοιάζει με τον αντίστοιχο αλγόριθμο για τους ακεραίους.

Λεπτομέρειες για αυτες τις πολυωνυμικές ακολουθίες υπολοίπων (polynomial remainder senquences ή prs) βρίσκονται στο άρθρο της Anna Johnson η οποία θεμελείωσε την θεωρία των ακολουθιών αυτών.

2 Πρώτος αλγόριθμος υπολογισμού των prs με τριγωνοποιηση πινάκων

Εστω ότι θέλουμε να υπολογίσουμε τις Ευκλείδειες και τροποποιημένες

Ευκλείδειες ακολουθίες για τα πολυώνυμα $f(x)=x^3+3\,x^2-7\,x+7$ και $g(x)=f'(x)=3\,x^2+6\,x-7$.

```
>>> from sympy.polys.subresultants_qq_zz import *
>>> from sympy import symbols, rem, pprint, Matrix
>>> x = symbols('x')
>>> f = x**3 + 3*x**2 - 7*x +7;
>>> g = 3*x**2 + 6*x - 7;
>>>
```

Η Ευκλείδεια prs των f, g είναι:

```
>>> euclid_amv(f, g, x)
```

```
[x**3 + 3*x**2 - 7*x + 7, 3*x**2 + 6*x - 7, 84 - 60*x,
2912]
```

```
>>>
```

ενώ η τροποποιημένη Ευκλείδεια prs των f, g είναι:

```
>>> sturm_amv(f, g, x)
```

```
[x**3 + 3*x**2 - 7*x + 7, 3*x**2 + 6*x - 7, 60*x - 84,
-2912]
```

```
>>>
```

διαφέρουν δηλαδή ως προς τα πρόσημα. Το πρώτο υπολοιπο της Ευκλείδειας prs είναι $-60x + 84$ το οποίο προκύπτει είτε από την συνάρτηση rem()

```
>>> rem(3**2 *f, g, x)
```

```
84 - 60*x
```

είτε από την συνάρτηση rem_z().

```
>>> rem_z(f, g, x)
```

```
84 - 60*x
```

Εμείς θα υπολογίσουμε το υπόλοιπο αυτό με τριγωνοποίηση του πίνακα

m, οπου

```
>>> m = Matrix([[3, 6, -7, 0], [0, 3, 6, -7], [1, 3, -7,
    7]])
>>> pprint(m)
```

```
[3  6  -7  0 ]
[            ]
[0  3  6   -7]
[            ]
[1  3  -7  7 ]
```

Προσέξτε πως στον πίνακα m στην τελευταία σειρά βρίσκονται οι συντελεστές του f ενώ στις άλλες 2 είναι οι συντελεστές του g. Γενικά, όταν defree$(f,x) >$ degree(g,x), οι διαστάσεις του πινακα m ειναι

$$(\text{degree}(f,x) - \text{degree}(g,x) + 2) \times (\text{degree}(f,x) + 1).$$

Στο sympy για την τριγωνοποίηση πινάκων υπάρχει μονο η συνάρτηση rref() η οποία δεν μας κάνει γιατί

a) μηδενίζει τα στοιχεία του πίνακα τόσο κάτω απο τον οδηγό (pivot) όσο και πάνω από αυτόν, και

b) οι πράξεις δεν γίνονται στους ακεραίους — όπως εμείς θέλουμε — αλλά στους ρητούς .

```
>>> m.rref()
```

```
(Matrix([
[1, 0, 0, -21/5],
[0, 1, 0,  7/15],
[0, 0, 1,  -7/5]]), (0, 1, 2))
```

Ετσι καταφεύγουμε στο Xcas που έχει την συνάρτηση pivot(), στην οποία το τέταρτο όρισμα (όταν υπάρχει, πχ $-i$) ειναι αρνητικό και σημαίνει πως δεν πειράζουμε τις σειρές πάνω απο την i.

```
> m := [ [3,6,-7,0], [0,3,6,-7], [1,3,-7,7] ]
```

$$\begin{bmatrix} 3 & 6 & -7 & 0 \\ 0 & 3 & 6 & -7 \\ 1 & 3 & -7 & 7 \end{bmatrix}$$

```
> m1 := pivot(m, 0, 0)
```

$$\begin{bmatrix} 3 & 6 & -7 & 0 \\ 0 & 3 & 6 & -7 \\ 0 & 3 & -14 & 21 \end{bmatrix}$$

```
> m2 := pivot(m1, 1, 1, -1)
```

$$\begin{bmatrix} 3 & 6 & -7 & 0 \\ 0 & 3 & 6 & -7 \\ 0 & 0 & -60 & 84 \end{bmatrix}$$

Βλέπουμε πως τριγωνοποιώντας τον πίνακα m βρίσκουμε το υπόλοιπο με συντελεστές -60 και 84 στον πινακα $m2$. Προφανώς το τροποποιημένο υπόλοιπο έχει συντελεστές 60 και -84.

Να προγραμματίσετε στο sympy την συνάρτηση pivot() του Xcas και με την βοήθειά της την συνάρτηση euclid_triang(f, g, x) που θα υπολογίζει την prs των f, g με τριγωνοποίηση πινάκων. Να υπολογίσετε την Ευκλείδεια ακολουθία για τα πολυώνυμα $[f(x) = x^3 + 3x^2 - 7x + 7,\ g(x) = 3\ x^2 + 6\ x - 7]$ και $[f = x^4 - x^3 + x^2 - 7\ x + 7,\ g = 4\,x^3 - 3\,x^2 + 2\,x - 7]$ και να συγκρίνετε τα αποτελέσματα σας με αυτά που προκύπτουν απο την συνάρτηση euclid_amv(f, g, x). Πώς συγκρίνονται οι συντελεστές τωνς πολυωνύμων και που οφείλεται η διαφορά τους;

Απάντηση 1.

```
Python 2.7.12 (default, Oct  8 2019, 14:14:10)
[GCC 5.4.0 20160609]
Python plugin for TeXmacs.
Please see the documentation in Help -> Plugins ->
Python
```

```
>>> from sympy.polys.subresultants_qq_zz import *
    from sympy.polys import *
    from sympy import symbols, rem, pprint, Matrix, zeros
    from sympy.matrices.matrices import  *



    def euclid_triang(f, g, x):

        #vathmos polyonimou
        d = degree(f, x)
        euclid_t_list = []
        euclid_t_list.append(f)
        euclid_t_list.append(g)
        m1 = poltoMatrix(f,g)

        endpoint = len(m1)

        while endpoint > 9 :

            mPivot = my_pivot(m1, 0, 0)
            mPivot2 = my_pivot(mPivot, 1, 1)
            d = ZZ.quo(len(mPivot2), 3)
            prs = rowToPol(mPivot2,d,2)
            euclid_t_list.append(prs)

            if len(mPivot2) > 9:
                fnew = rowToPol(mPivot2,d,1)
                gnew = rowToPol(mPivot2,d,2)
                d = d - 1
                m1 = poltoMatrix(fnew,gnew)

            endpoint = len(mPivot2)

        print(euclid_t_list)
```

```
>>> def my_pivot(m, row, col):

        d = ZZ.quo(len(m), 3)
        x = symbols('x')
        firstRow = rowToPol(m,d,0)
        middleRow = rowToPol(m,d,1)
        lastRow = rowToPol(m,d,2)
        if row == 0 and col ==0 :
            rmnt = rem_z(lastRow, firstRow, x)
        else :
            rmnt = rem_z(lastRow, middleRow, x)
        m = poltoMatrix(rmnt,middleRow)
        return(m)
     #metatrepei poly se Matrix
    def poltoMatrix(f,g):

        df = degree(f, x)
        dg = degree(g, x)
        #dhmiourgia pinakon me stoixeia polyonimon
        fM = Matrix(1, df+1, Poly(f, x).all_coeffs())
        gM = Matrix(1, dg+1, Poly(g, x).all_coeffs())

        #dimioyrgia g vathmou f
        mG = zeros(1,1)
        mG1 = gM.col_insert(dg+1,mG)
        mG2 = gM.col_insert(0,mG)
        m1 = mG1
        m1 = m1.row_insert(1, mG2)

        if df == dg:

            fM=fM.col_insert(0,mG)

        if df < dg:
            for i in range(2):
                fM=fM.col_insert(0,mG)
                i = i + 1
            i = 0
        m1 = m1.row_insert(2, fM)
        return(m1)
```

```
>>>

>>> def rowToPol(m,d,row):#dexetai pinaka epistrefei poly
        i=0;gN="";last = 0
        while i < d:
            if i != d-1 :
                if i != d-2:
                    if m[row, i] < 0:
                        s = str(-1*m[row,
    i])+"*x**"+str(d-i-1);pros = -1
                    else:
                        s = str(m[row,
    i])+"*x**"+str(d-i-1);pros = 1
                else :
                    if m[row, i] < 0:
                        s = str(-1*m[row, i])+"*x";pros =
    -1
                    else:
                        s = str(m[row, i])+"*x";pros = 1
                i = i+1
            else :
                if m[row, i] < 0:
                    s = str(-1*m[row, i]);pros = -1;last =
    1
                else:
                    s = str(m[row, i]);pros = 1;last = 1
                if pros == -1:
                    gN = gN +" - "+ s
                else:
                    gN = gN +" + "+ s
                i=i+1
            if i == 1 :
                if pros == -1:
                    gN = " - "+ s
                else:
                    gN = s
            else:
                if last != 1 :
                    if pros == -1:
                        gN = gN +" - "+ s
                    else:
                        gN = gN +" + "+ s
        f = eval(gN);return f
```

```
>>>
>>> euclid_triang(f, g, x)
    [x**3 + 3*x**2 - 7*x + 7, 3*x**2 + 6*x - 7, 84 - 60*x,
    26208]
>>> euclid_amv(f, g, x)
    [x**3 + 3*x**2 - 7*x + 7, 3*x**2 + 6*x - 7, 84 - 60*x,
    2912]
>>>
>>> f = x**4 - x**3 + x**2 - 7*x +7;
>>> g = 4*x**3 - 3*x**2 + 2*x -7;
>>> euclid_triang(f, g, x)
    [x**4 - x**3 + x**2 - 7*x + 7, 4*x**3 - 3*x**2 + 2*x -
    7, 5*x**2 - 82*x + 105, 23616*x - 33040, 35974400]
>>> euclid_amv(f, g, x)
    [x**4 - x**3 + x**2 - 7*x + 7, 4*x**3 - 3*x**2 + 2*x -
    7, 5*x**2 - 82*x + 105, 1476*x - 2065, 5621]
>>>
```

Σχόλιο 1. Οι συνελεστές δεν έχουν διαιρεθεί με το β (coefficients-reduction factor) γι αυτό και είναι τόσο μεγαλύτεροι σε σχέση με αυτούς που προκύπτουν από euclid_amv.

3 Δεύτερος αλγόριθμος υπολογισμού των prs με τους πίνακες Sylvester

Στην ενότητα αυτή θα υπολογίσουμε τις Ευκλείδειες και τροποποιημένες Ευκλείδειες ακολουθίες με την βοήθεια των πινάκων sylvester(f, g, x, 1) και sylvester(f, g, x, 2) αντιστοιχα . Και στις δύο περιπτώσεις θα χρησιμοποιήσουμε τα πολυώνυμα της προηγούμενης ενώτητας, δηλ. $f = x^3 + 3\,x^2 - 7\,x + 7$ και $g = 3\,x^2 + 6\,x - 7$.

3.1 Ευκλείδειες ακολουθίες και sylvester(f, g, x, 1)

Για τα πολυώνυμα του παραδείγματός μας ο πίνακας sylvester(f, g, x, 1) είναι:

```
>>> s1 = sylvester(f, g, x, 1) ; pprint( s1 )

[1  3  -7  7   0 ]
[                ]
[0  1  3   -7  7 ]
[                ]
[3  6  -7  0   0 ]
[                ]
[0  3  6   -7  0 ]
[                ]
[0  0  3   6   -7]
```

Το υπόλοιπο που θέλουμε να υπολογίσουμε κατά πάσα πιθανότητα θα είναι βαθμού 1. Για να βρούμε τους 2 συντελεστές του, απαλείφουμε την τελευταία σειρά των συντελέστών του f και την τελευταία σειρά των συντελεστών του g για να προκύψει ενας 3×4 πίνακας (η τελευταία στήλη είναι μηδενική και την αγνοούμε). Από αυτόν τον πίνακα σχηματίζουμε δύο 3×3 πίνακες και οι ορίζουσές των ειναι οι ζητούμενοι συντελεστές.

```
>>> s1.row_del(4); s1.row_del(1); print(s1[:, 0:3].det())

-60

>>> s1.col_swap(2, 3); print(s1[:, 0:3].det())

84
```

Το τελευταίο πολυώνυμο της ακολουθίας είναι μηδενικού βαθμού (σταθερά) και στην περίπτωση αυτή παίρνουμε την ορίζουσα του αρχικού πίνακα s1 (δεν απαλείφουμε καμία σειρά από τον πίνακα). Βλέπε και σελ. 22–23 του άρθρου για την Anna Johnson.

Να προγραμματίσετε στο sympy την συνάρτηση euclid_sylv1(f, g, x) που θα υπολογίζει την Ευκλείδεια prs των f, g. Να υπολογίσετε την Ευκλείδεια ακολουθία για τα πολυώνυμα $[f(x) = x^3 + 3x^2 - 7x + 7$, $g(x) = 3\,x^2 + 6\,x - 7]$ και $[f(x) = x^8 + x^6 - 3\,x^4 - 3\,x^3 + 8\,x^2 + 2\,x - 5$, $g(x) = 3\,x^6 + 5\,x^4 - 4\,x^2 - 9\,x + 21]$ και να συγκρίνετε τα πρόσημα των αποτελεσμάτων σας με αυτά που προκύπτουν από την συνάρτηση euclid_amv(f, g, x).

Απάντηση 2.

3.2 Τροποποιημένες Ευκλείδειες ακολουθίες και sylvester(f, g, x, 2)

Για τα πολυώνυμα του παραδείγματός μας ο πίνακας sylvester(f, g, x, 2) είναι :

```
>>> s2 = sylvester(f, g, x, 2) ; pprint( s2)
```

```
[1  3  -7  7   0   0 ]
[                    ]
[0  3  6   -7  0   0 ]
[                    ]
[0  1  3   -7  7   0 ]
[                    ]
[0  0  3   6   -7  0 ]
[                    ]
[0  0  1   3   -7  7 ]
[                    ]
[0  0  0   3   6   -7]
```

Οπως και πριν, το υπόλοιπο που θέλουμε να υπολογίσουμε κατά πάσα πιθανότητα θα είναι βαθμού 1. Για να βρούμε τους 2 συντελεστές του, απαλείφουμε το *τελευταίο ζεύγος* των συντελεστών των f και g για να προκύψει ένας 4×5 πίνακας (η τελευταία στήλη είναι μηδενική και την αγνοούμε). Από αυτόν τον πίνακα σχηματίζουμε δύο 4×4 πίνακες και οι ορίζουσες των είναι οι ζητούμενοι συντελεστές.

```
>>> s2.row_del(4); s2.row_del(4); print(s2[:, 0:4].det())
```

```
60
```

```
>>> s2.col_swap(3, 4); print(s2[:, 0:4].det())
```

```
-84
```

Βλέπε και σελ. 24 – 25 του άρθρου για την Anna Johnson.

Να προγραμματίσετε στο sympy την συνάρτηση sturm_sylv2(f, g, x) που θα υπολογίζει την τροποποιημένη Ευκλείδεια prs των f, g. Να υπολογίσετε την τροποποιημένη Ευκλείδια ακολουθία για τα πολυώνυμα $[f(x) = x^3 + 3\,x^2 - 7\,x + 7,\ g(x) = 3\,x^2 + 6\,x - 7]$ και $[f(x) = x^8 + x^6 - 3\,x^4 - 3\,x^3 + 8\,x^2 + 2\,x - 5,\ g(x) = 3\,x^6 + 5\,x^4 - 4\,x^2 - 9\,x + 21]$ και να συγκρίνετε τα πρόσημα των αποτελεσμάτων σας με αυτά που προκύπτουν απο την συνάρτηση sturm_amv(f, g, x).

Απάντηση 3.

4 Το θεώρημα της Anna Johnson και η σημασια του

Οπως είδαμε στην προηγούμενη ενότητα για $[f(x)=x^8+x^6-3\,x^4-3\,x^3+8\,x^2+2\,x-5, g(x)=3\,x^6+5\,x^4-4\,x^2-9\,x+21]$ τα πολυώνυμα των euclid_sylv1(f, g, x) και sturm_sylv2(f, g, x) διαφέρουν στα πρόσημα από τα αντίστοιχα πολυώνυμα των euclid_amv(f, g, x) και sturm_amv(f, g, x).

Προσέξτε πως τα πολυώνυμα των euclid_sylv1(f, g, x) και sturm_sylv2(f, g, x) έχουν πάντα *ακέραιους συντελεστές* επειδή υπολογίζονται απο ορίζουσες πινάκων με ακέραια στοιχεία. Αντίθετα τα πολυώνυμα με euclid_amv(f, g, x) και sturm_amv(f, g, x) υπολογίζονται με διαιρέσεις και (χρησιμοποιώντας διαφορετικές συναρτήσεις) μπορούν να είναι και ρητοί αριθμοί. Ετσι έχουμε:

```
>>> f = x**8 + x**6 - 3*x**4 - 3*x**3 + 8*x**2 + 2*x - 5
>>> g = 3*x**6 + 5*x**4 - 4*x**2 - 9*x +21
>>> euclid_q(f, g, x)

  [x**8 + x**6 - 3*x**4 - 3*x**3 + 8*x**2 + 2*x - 5,
  3*x**6 + 5*x**4 - 4*x**2 - 9*x + 21, -5*x**4/9 + x**2/9
  - 1/3, -117*x**2/25 - 9*x + 441/25, 233150*x/19773 -
  102500/6591, -1288744821/543589225]

>>> sturm_q(f, g, x)

  [x**8 + x**6 - 3*x**4 - 3*x**3 + 8*x**2 + 2*x - 5,
  3*x**6 + 5*x**4 - 4*x**2 - 9*x + 21, 5*x**4/9 - x**2/9
  + 1/3, 117*x**2/25 + 9*x - 441/25, 233150*x/19773 -
  102500/6591, -1288744821/543589225]

>>>
```

Ερώτηση 1. Παρατηρώντας τα παραπάνω αποτελέσματα των euclid_q(f, g, x) και strum_q(f, g, x) βρείτε την σχέση μεταξύ των προσήμων ανάμεσα στις Ευκλείδειες και τις τροποποιημένες Ευκλείδειες ακολουθίες.

```
>>> euclid_sylv1(f, g, x)
```

```
>>> sturm_sylv2(f, g, x)
```

Απάντηση 4.

Ερώτηση 2. Η απάντηση που δώσατε στην 1η Ερώτηση ισχύει για τα παραπάνω αποτελέσματα των euclid_sylv1(f, g, x) και sturm_sylv2(f, g, x);

```
>>> sturm_q(f, g, x)
    [x**8 + x**6 - 3*x**4 - 3*x**3 + 8*x**2 + 2*x - 5,
    3*x**6 + 5*x**4 - 4*x**2 - 9*x + 21, 5*x**4/9 - x**2/9
    + 1/3, 117*x**2/25 + 9*x - 441/25, 233150*x/19773 -
    102500/6591, -1288744821/543589225]
>>> sturm_sylv2(f, g, x)
```

Απάντηση 5.

Ερώτηση 3. Μπορείτε να βρείτε πώς σχετίζονται οι συντελεστές του πολυωνύμου $\frac{117x^2}{25} + 9\ x - \frac{441}{25}$ της ακολουθίας sturm_q(f, g, x) — υπολογισμοί με διαιρέσεις — με τους αντίστοιχους συντελεστές του πολυωνύμου $65\ x^2 + 125\ x - 245$ της ακολουθίας sturm_sylv2(f, g, x), όπου οι συντελεστές ειναι ορίζουσες;

Απάντηση 6.

Την απάντηση στην Ερώτηση 3. έδωσε η Anna Johnson στο θεμελειώδες θεώρημά της του 1917. Να γράψετε τον τύπο (5) της σελίδας 30 του άρθρου της, από τον οποίο προκύπτει πως οι συντελεστές του πολυωνύμου $\frac{117x^2}{25} + 9\,x - \frac{441}{25}$ υπολογίζονται από τους συντελεστές του πολυωνύμου $65\,x^2 + 125\,x - 245$ αν πολλαπλασιάσουμε τους τελευταίους με το $\frac{9}{125}$, δηλαδή $\frac{9}{125} \cdot 65 = \frac{117}{25}$, κοκ. Τα 4 υπόλοιπα συνδεεονται μεταξυ τους με τους ρητούς

$$-\frac{1}{27}, \frac{9}{125}, -\frac{25}{19773}, -\frac{19773}{2174356900}.$$

Τύπος (5) σελίδας 30.

- Με την βοήθεια των ανωτέρω από το πολυώνυμο $-15\,x^4 + 3\,x^2 - 9$ της ακολουθίας sturm_sylv2(f, g, x) να υπολογίσετε το αντίστοιχο πολυώνυμο της ακολουθίας sturm_q(f, g, x).
- Οπως βλέπετε παρακάτω, τα πολυώνυμα $R^{(i)}$ στην ακολουθία sturm_amv(f, g, x) έχουν ακέραιους συντελεστές $r_k^{(i)}$.

```
>>> sturm_amv(f, g, x)
    [x**8 + x**6 - 3*x**4 - 3*x**3 + 8*x**2 + 2*x -
    5, 3*x**6 + 5*x**4 - 4*x**2 - 9*x + 21, 15*x**4 -
    3*x**2 + 9, 65*x**2 + 125*x - 245, 9326*x -
    12300, -260708]
>>> sturm_sylv2(f, g, x)
```

5 Ακολουθίες Sturm για την απομόνωση των πραγματικών ριζών πολυωνύμων με ακέραιους συντελεστες

Ακολουθίες Sturm ονομάζονται οι τροποποιημένες Ευκλείδειες ακολουθίες των πολυωνύμων f, g στην ειδική περίπτωση που $g = f'$. Οι ακολουθίες αυτές μας χρησιμεύουν στην απομόνωση των πραγματικών ριζών πολυωνύμων με ακέραιους συντελεστές. Για ευκολία θα χρησιμοποιήσουμε την συνάρηση sturm(f, f', x) του sympy.

Ο Sturm (1829) εμπνευστηκε το θεώρημα του (και την μέθοδό του) από την εργασία του Fourier (1820). Περισσότερες λεπτομέρειες πάνω στο Θεώρημα Fourier και την μέθοδο Sturm βρίσκονται στις Ελληνικές σημειώσεις (2005) σελ 7-43.

Για να προγραμματίσουμε την μέθοδο του Sturm για την απομόνωση των θετικών ριζών ενός πολυωνύμου (μέθοδος διχοτόμησης ενός διαστήματος) χρειαζόμαστε τις εξής συναρτήσεις:

i. sturm(f), απο το sympy, για τον υπολογισμό της ακολουθίας,

ii. cauchy_upper_bound(f, x) για την εύρεση ενός αρχικού διαστήματος μέσα στο οποίο βρίσκονται οι θετικές ρίζες,

iii. sign_var(num_list) για την εύρεση των μεταβολών προσήμου μιας αριθμητικής ακολουθίας, με στοιχεία διάφορα του μηδενός,

iv. square_free_factorts(f, x), για την εύρεση πολυωνύμων χωρίς πολλαπλές ριζες.

Οι αρνητικές ρίζες απομονώνονται αφού αντικαταστήσουμε την μεταβλητή x με $-x$. Ο αλγόριθμος περιγράφεται στην σελ. 30 των σημειώσεων.

Να απομονώσετε τις πραγματικές ρίζες των πολυωνύμων $x^3 - 7\,x + 7$ και $32\,x^6 - 48\,x^4 + 18\,x^2 - 1$ και να τις προσεγγίσετε με ακρίβεια 7 ψηφίων χρησιμοποιώντας την συνάρτηση refine_root(f, b, c, eps = 1e-7) του sympy

Απάντηση 7.