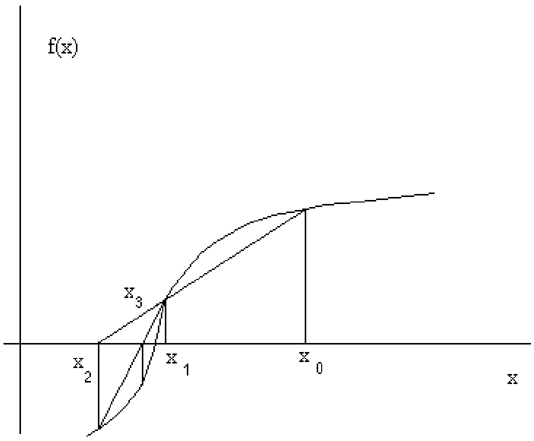
ΤΕΜΦΕ 4^o Εξάμηνο
Αριθμητική Ανάλυση Ι
1^η Εργαστηριακή Άσκηση

Μέθοδος Müller

Αν θέλαμε να ερμηνεύσουμε γεωμετρικά τη μέθοδο Secant θα βλέπαμε ότι σε κάθε βήμα φέρουμε την ευθεία που διέρχονται από τις εικόνες δύο σημείων π.χ. x_0 και x_1. Το επόμενο σημείο της προσέγγισης x_2 είναι η τομή της ευθείας με τον άξονα των x. Η διαδικασία επαναλαμβάνεται για τα σημεία x_1 και x_2.



Αν αντί για δύο δοθέντα σημεία σε κάθε βήμα είχαμε τρία, π.χ. x_0, x_1 και x_2, τότε θα μπορούσαμε να φέρουμε από τις εικόνες τους μία παραβολή. Το επόμενο σημείο της προσέγγισης (το x_3) είναι η τομή της παραβολής με τον άξονα των x. Στο επόμενο βήμα, η διαδικασία επαναλαμβάνεται για τα σημεία x_1, x_2 και x_3.

Η μέθοδος που περιγράψαμε ονομάζεται μέθοδος του Müller και ο επαναληπτικός της τύπος δίνεται από τους ακόλουθους τύπους:

$$c = f(x_2)$$

$$b = \frac{(x_0-x_2)^2\left[f(x_1)-f(x_2)\right]-(x_1-x_2)^2\left[f(x_0)-f(x_2)\right]}{(x_0-x_2)(x_1-x_2)(x_0-x_1)}$$

$$a = \frac{(x_1-x_2)\left[f(x_0)-f(x_2)\right]-(x_0-x_2)\left[f(x_1)-f(x_2)\right]}{(x_0-x_2)(x_1-x_2)(x_0-x_1)}$$

$$x_3 = x_2 - \frac{2c}{b \pm \sqrt{b^2-4ac}}$$

ενώ η εξίσωση της παραβολής είναι η

$$p(x) = a(x-x_2)^2 + b(x-x_2) + c\,.$$

Η παραβολή μπορεί να τέμνει τον άξονα των x σε δύο σημεία, οπότε υπάρχουν δύο επιλογές για το x_3. Επιλέγουμε ως νέα προσέγγιση το σημείο τομής που είναι πιο κοντά στην προηγούμενη με την κατάλληλη επιλογή του προσήμου στον παρανομαστή. Στον τύπο θέτουμε + ή – ανάλογα ώστε ο παρανομαστής να είναι έχει το μεγαλύτερο δυνατό μέτρο.

Σας δίνεται η ακόλουθη πολυωνυμική εξίσωση

$$x^4 + 2x^3 + (-3a - \frac{5}{4})x^2 + 2a^2x - a^4 - 3a^3 - \frac{5a^2}{4} = 0$$

όπου το a ισούται με το τελευταίο ψηφίο του αριθμού μητρώου σας αυξημένο κατά ένα. Για την εξίσωση αυτή γνωρίζουμε ότι έχει δύο πραγματικές ρίζες στο διάστημα $\left[-a-5, a+5\right]$και ένα ζεύγος μιγαδικών ριζών.

Ερωτήματα

1. Εκμεταλλευόμενοι τις δυνατότητες γραφικών του MATLAB, προσδιορίστε δύο διαστήματα (με πλάτος από 0.8 έως 1.2) που το καθένα να περιέχει μία από τις δύο πραγματικές ρίζες.
2. Χρησιμοποιώντας ως αρχικές τιμές τα άκρα του κάθε διαστήματος για τη Secant και ένα τρίτο εσωτερικό σημείο για την Müller προσεγγίστε αυτές τις πραγματικές ρίζες και με τις δύο μεθόδους.
3. Με βάση τα αποτελέσματα συγκρίνετε τις δύο μεθόδους. Ποια από τις δύο εμφανίζει να έχει μεγαλύτερη τάξη σύγκλισης.
4. Επιλέγοντας ως αρχικές τιμές άκρα διαστήματος που περιέχουν την αρχή των αξόνων για τη Secant και ένα τρίτο εσωτερικό σημείο για την Müller, προσπαθήστε να προσεγγίσετε τη μιγαδική ρίζα και με τις δύο μεθόδους. Σχολιάστε τα αποτελέσματα σας.

Κώδικας

Η μέθοδος του Müller αν προγραμματιστεί κατάλληλα μπορεί να προσεγγίσει και μιγαδικές ρίζες και σε αυτή την περίπτωση κώδικας της σε MATLAB είναι ο ακόλουθος:

```
function [x,xlist,iter] = muler(f,x0,x1,x2,tol,maxit)
% Root of f(x)=0 by muler method.
% method terminates when relative change > tol
% or maximum number exceeds of iterations
if nargin < 4,
   fprintf('Insufficient input \n');     break;
end;
if nargin < 5,tol = eps; end;
if nargin < 6, maxit = 50;  end;
f0 = feval(f,x0); f1 = feval(f,x1);
iter = 0; xdiff = inf;
xlist=[x0;x1;x2];
while xdiff >= tol
   f2 = feval(f,x2);
   c0=(f1-f0)/(x1-x0);
   c1=(f2-f1)/(x2-x1);
   d0=(c1-c0)/(x2-x0);
   s=c1+d0*(x2-x1);
   dd=sqrt(s^2-4*f2*d0);
   if (abs(s-dd)<abs(s+dd)),
      ss=1;
   else
      ss=-1;
   end;
   x = x2 - 2*f2/(s+ss*dd);
   xdiff = abs(x-x2)/abs(x);
   xlist=[xlist;x];
   iter = iter + 1;
   if iter >= maxit
      disp('Not converged after maxit iterations.');
      break;
   end
   x0=x1;f0=f1;
   x1=x2;f1=f2;
   x2=x;
end
```

Στον κώδικα το a είναι το d0, το b το s και το c το f2, και κατά την κλίση της συνάρτησης το x_2 επιλέγεται να βρίσκεται ανάμεσα στα x_0 και x_1.

Οδηγίες

- Οι κλήσεις των μεθόδων θα γίνουν με τιμή της παραμέτρου ανοχής TOL ίση με 10^{-7}.
- Η εργασία θα πρέπει να παραδοθεί κατά τη διάρκεια του 4^{ου} εργαστηρίου.
- Θα πρέπει να παραδοθεί εκτυπωμένη εργασία συρραμμένη (απλά) έτσι ώστε να μπορεί κάποιος να την ξεφυλλίσει. Η εργασία θα πρέπει να έχει **εξώφυλλο** στο οποίο να αναφέρεται ο αύξων αριθμός της, το όνομα και ο αριθμός μητρώου του φοιτητή, τα στοιχεία της σχολής και του μαθήματος, η ημερομηνία παράδοσης και το τμήμα το οποίο παρακολουθεί εργαστήριο. Σε **παράρτημα** θα πρέπει να υπάρχουν εκτυπωμένοι οι κώδικες (τα προγράμματα, scripts και .m αρχεία). Στο κύριο μέρος της εργασίας θα πρέπει να αναπτύσσονται η διαδικασία, τα σχόλια, τα γραφήματα **και μόνο όσα από τα αποτελέσματα είναι απαραίτητα για τα συμπεράσματα**. Όλα αυτά θα πρέπει να έχουν τη συνοχή ενιαίου κειμένου. Εκτός από τις όποιες εκτυπώσεις θα πρέπει να παραδοθούν τα προγράμματα και τα αποτελέσματα τους **σε ηλεκτρονική μορφή (δισκέτα)**. Τα αποτελέσματα μπορείτε να τα αποθηκεύσετε με τη χρήση της diary.
- Αν δεν υπάρχει δυνατότητα παράδοσης της εργασίας σε έντυπη (εκτυπωμένη) μορφή, **γίνεται δεκτή και δισκέτα** που να περιέχει όλα όσα αναφέρονται παραπάνω. Δηλαδή, επιπλέον των προγραμμάτων και αποτελεσμάτων η δισκέτα θα πρέπει να περιέχει και **ένα αρχείο Word** που θα έχει ως περιεχόμενο **όλα όσα θα παραδίδατε σε εκτυπωμένη μορφή**.
- **Τόσο η πληρότητα, τα σχόλια το αν ακολουθήθηκαν οι οδηγίες αλλά και ο τρόπος της παρουσίασης των αποτελεσμάτων της εργασίας** που θα παραδοθεί θα ληφθούν υπ' όψιν κατά την αξιολόγηση.

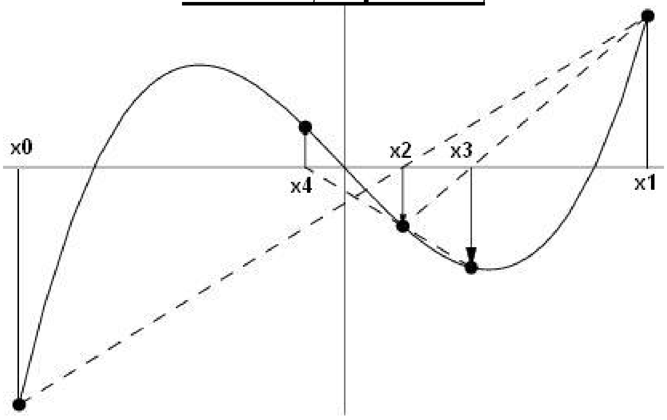
ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΤΜΗΜΑ ΕΦΑΡΜΟΣΜΕΝΩΝ ΜΑΘΗΜΑΤΙΚΩΝ ΚΑΙ ΦΥΣΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΙΚΑΡΙΟΣ ΕΜΜΑΝΟΥΗΛ
Α.Μ. 09101128
4ο ΕΞΑΜΗΝΟ

ΑΡΙΘΜΗΤΙΚΗ ΑΝΑΛΥΣΗ Ι
1η ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ
ΕΠΙΛΥΣΗ ΜΗ ΓΡΑΜΜΙΚΩΝ ΕΞΙΣΩΣΕΩΝ
ΤΜΗΜΑ: ΠΑ1Α
ΗΜ/ΝΙΑ ΠΑΡΑΔΟΣΗΣ
5/6/2003

Μέθοδος Τέμνουσας



ΕΡΩΤΗΜΑ 1

Για να προσδιορίσουμε τα ζητούμενα διαστήματα πρέπει καταρχάς να δημιουργήσουμε τη γραφική παράσταση της δοσμένης εξίσωσης:

$$x^4 + 2x^3 + (-3 \cdot 9 - \frac{5}{4})x^2 + 2 \cdot 9^2 x - 9^4 - 3 \cdot 9^3 - \frac{5 \cdot 9^2}{4} = 0$$

όπου α=8+1=9, αφού ο αριθμός μητρώου μου είναι 0910112***8***

Δημιουργούμε λοιπόν δύο m-files τα myequation.m και question1.m τα οποία περιέχουν τα εξής:

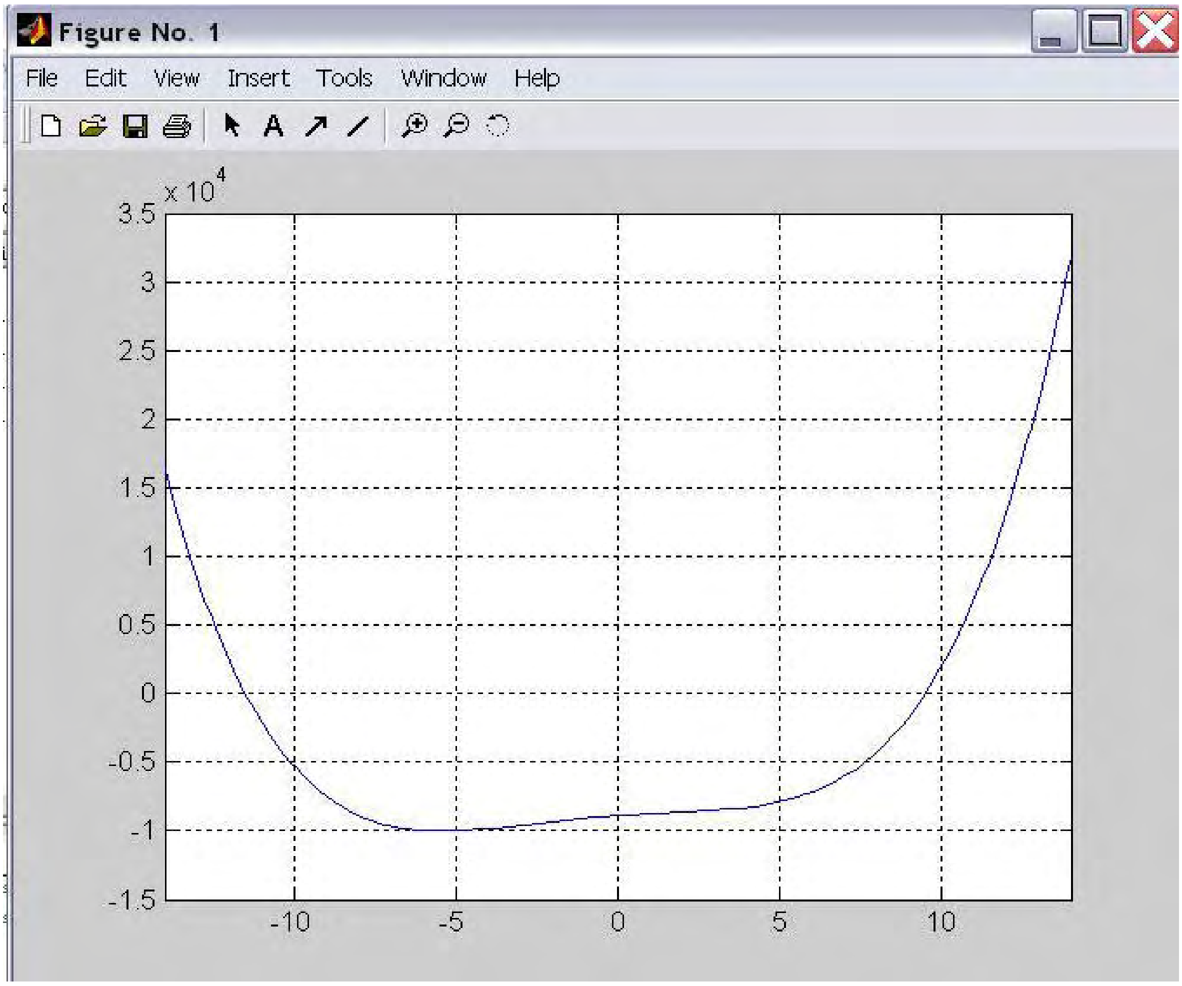
myequation.m	***question1.m***

function y=myequation(x);
%This is the given equation, where a=9
%Since my number is 09101128
%And a=8+1=9
y=x.^4+2*x^3+(-3*9-5/4)*x.^2+2*x*9^2-9^4-3*9^3-(5*9^2)/4;
	%This script shows the plot of myequation
	%In [-a-5,a+5],
	%When a=8+1=9
	fplot('myequation',[-14 14]);
	grid on;

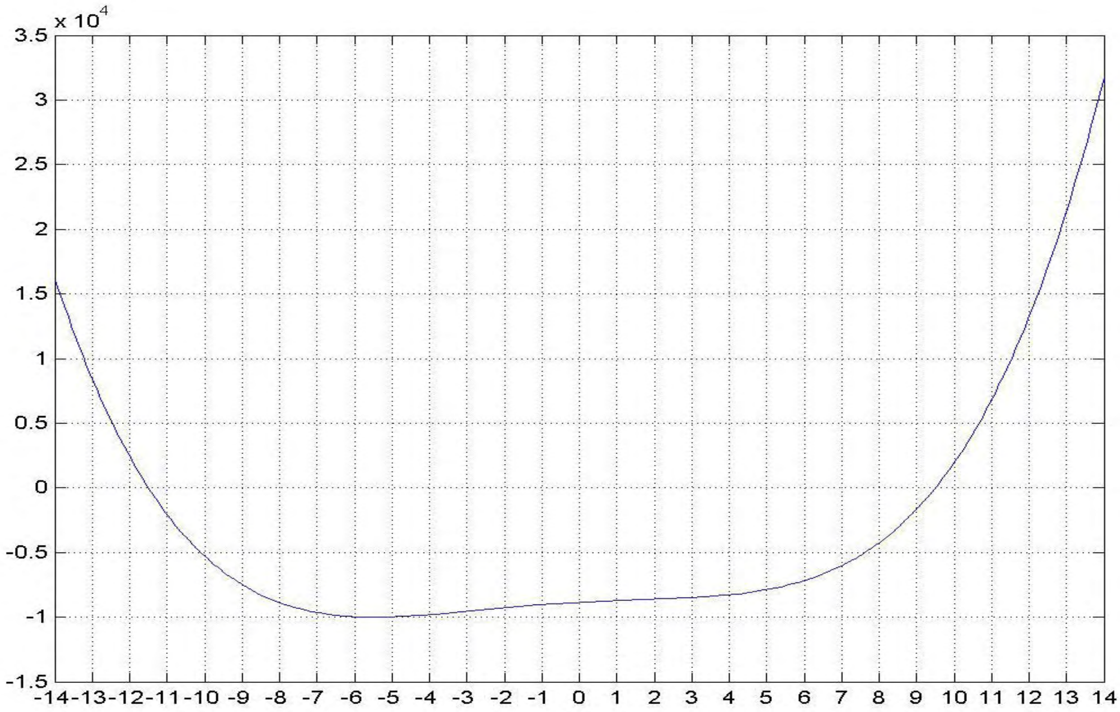
Εκτελώντας την εντολή

>>question1

Εμφανίζεται η γραφική παράσταση της συνάρτησης όπως αυτή φαίνεται παρακάτω:



Στη συνέχεια μέσω της επιλογής Axes Properties… και πληκτρολογώντας -14:1:10 στην επιλογή Ticks καταλήγουμε στην παρακάτω εικόνα:



Οπότε παρατηρούμε πως τα ζητούμενα διαστήματα είναι τα (-12,-11) και (9,10).

ΕΡΩΤΗΜΑ 2

Για να προσδιορίσουμε τις ρίζες στα προηγούμενα διαστήματα εκτελούμε τις παρακάτω εντολές:

```
>>format long
>>[x,l,iter]=secant('myequation',-12,-11,1e-7)
>>[x,l,iter]=muller('myequation',-12,-11.3,-11,1e-7)
>>[x,l,iter]=secant('myequation',9,10,1e-7)
>>[x,l,iter]=muller('myequation',9,9.3,10,1e-7)
```

ΕΡΩΤΗΜΑ 3

Τα αποτελέσματα της παραπάνω διαδικασίας φαίνονται στον παρακάτω πίνακα:

	Μέθοδος Secant	Μέθοδος Muller
(-12,-11)	-11.50000000000003	-11.49999999999888
Επαναλήψεις	5	3
(9,10)	9.49999999999997	9.49999999999808
Επαναλήψεις	5	3

Από τα παραπάνω αποτελέσματα καταλήγουμε στο συμπέρασμα ότι η μέθοδος ***Muller*** έχει μεγαλύτερη τάξη σύγκλισης, αφού προσέγγισε τη ρίζα σε λιγότερες επαναλήψεις από τη μέθοδο ***Secant***.

ΕΡΩΤΗΜΑ 4

Για να προσεγγίσουμε τη μιγαδική ρίζα εκτελούμε τις παρακάτω εντολές:

```
>>[x,l,iter]=secant('myequation',-3,3,1e-7)
>>[x,l,iter]=muller('myequation',-3,1,3,1e-7)
```

Οπότε τα αποτελέσματα που προκύπτουν είναι τα εξής:

	Μέθοδος Secant	Μέθοδος Muller
(-3,3)	9.50000000000000	-0.0000000000 + 9.0000000000i
Επαναλήψεις	39	9

Από τα παραπάνω παρατηρούμε ότι ενώ η μέθοδος ***Muller*** συγκλίνει στη μιγαδική ρίζα σε 9 μόλις επαναλήψεις, η μέθοδος ***Secant*** χρειάζεται 39 επαναλήψεις για να προσεγγίσει μία πραγματική ρίζα, η οποία είναι αυτή που βρέθηκε στο διάστημα (9,10).

ΠΑΡΑΡΤΗΜΑ

M-FILES

Muller.m

```
function [x,xlist,iter] = muler(f,x0,x1,x2,tol,maxit)
% Root of f(x)=0 by muler method.
% method terminates when relative change > tol
% or maximum number exceeds of iterations
if nargin < 4,
    fprintf('Insufficient input \n');
    break;
end;
if nargin < 5,tol = eps; end;
if nargin < 6, maxit = 50;  end;
f0 = feval(f,x0);
f1 = feval(f,x1);
iter = 0;
xdiff = inf;
xlist=[x0;x1;x2];
while xdiff >= tol
    f2 = feval(f,x2);
    c0=(f1-f0)/(x1-x0);
    c1=(f2-f1)/(x2-x1);
    d0=(c1-c0)/(x2-x0);
    s=c1+d0*(x2-x1);
    dd=sqrt(s^2-4*f2*d0);
    if (abs(s-dd)<abs(s+dd)),
        ss=1;
    else
        ss=-1;
    end;
    x = x2 - 2*f2/(s+ss*dd);
    xdiff = abs(x-x2)/abs(x);
    xlist=[xlist;x];
    iter = iter + 1;
    if iter >= maxit
       disp('Not converged after maxit iterations.');
       break;
   end
   x0=x1;f0=f1;
   x1=x2;f1=f2;
   x2=x;
end
```

secant.m

```
function [x,xlist,iter] = secant(f,x0,x1,tol,maxit)
% Root of f(x)=0 by secant method.
% method terminates when relative change > tol
% or maximum number exceeds of iterations
if nargin < 3,
    fprintf('Insufficient input \n');
    break;
end;
if nargin < 4,tol = eps; end;
if nargin < 5, maxit = 50;  end;
f0 = feval(f,x0);
f1 = feval(f,x1);
iter = 0;
xdiff = inf;
xlist=[x0;x1];
x=x1;
while xdiff >= tol
    xold=x;
    x = x - f1*(x1-x0)/(f1-f0);
    xdiff = abs(x-xold)/abs(x);
    xlist=[xlist;x];
    iter = iter + 1;
    if iter >= maxit
       disp('Not converged after maxit iterations.');
       break;
   end
   x0=x1;
   f0=f1;
   x1=x;
   f1 = feval(f,x);
end
```

myequation.m

```
function y=myequation(x);
%This is the given equation, where a=9
%Since my number is 09101128
%And a=8+1=9
y=x.^4+2*x^3+(-3*9-5/4)*x.^2+2*x*9^2-9^4-3*9^3-(5*9^2)/4;
```

question1.m

```
%This m-file shows the plot of myequation
%In [-a-5,a+5],
%When a=8+1=9
fplot('myequation',[-14 14]);
grid on;
```

DIARIES

question2.txt

```
diary('question2.txt')
format long

[x,l,iter]=secant('myequation',-11,-12,1e-7)

x =

 -11.49999999999998


l =

 -11.00000000000000
 -12.00000000000000
 -11.46121289747731
 -11.49710983497222
 -11.50001749600103
 -11.49999999213618
 -11.49999999999998


iter =

     5

[x,l,iter]=muller('myequation',-11,-11.3,-12,1e-7)

x =

 -11.49999999999883


l =

 -11.00000000000000
 -11.30000000000000
 -12.00000000000000
 -11.49951213149348
 -11.49999951787085
 -11.49999999999883


iter =

     3
```

```
[x,l,iter]=secant('myequation',9,10,1e-7)

x =

   9.49999999999997


l =

   9.00000000000000
  10.00000000000000
   9.46045060658579
   9.49698948167109
   9.50001895099262
   9.49999999095094
   9.49999999999997


iter =

     5

[x,l,iter]=muller('myequation',9,9.3,10,1e-7)

x =

   9.49999999999808


l =

   9.00000000000000
   9.30000000000000
  10.00000000000000
   9.49944813273462
   9.49999938227284
   9.49999999999808


iter =

     3

diary off
```

question3.txt

```
[x,l,iter]=secant('myequation',-3,3,1e-7)

x =

   9.50000000000000


l =

  -3.00000000000000
   3.00000000000000
  50.12500000000000
   3.06148498243490
   3.12281984807900
  55.06225992023209
   3.16933905485169
   3.21577366384870
  52.05850708615741
   3.27035979714629
   3.32482216476081
  49.03160626319711
   3.38950351926935
   3.45399863465454
  45.63470015596179
   3.53319279523239
   3.61208326936777
  41.79603435263118
   3.71337182476940
   3.81410859745665
  37.40867848838595
   3.95185712375645
   4.08843609029767
  32.33710212799699
   4.29338094742349
   4.49520560707856
  26.44007924579064
   4.84418966613173
   5.18126105748794
  19.73072963695654
   5.89654114127367
   6.53389412440731
  13.11278442091412
   8.02927419084958
   8.82566383746308
   9.69270575892132
   9.47852215474366
   9.49935357426670
   9.50000220584295
```

```
   9.49999999977388
   9.50000000000000


iter =

    39

[x,l,iter]=muller('myequation',-3,1,3,1e-7)

x =

 -0.00000000000000 + 9.00000000000000i


l =

 -3.00000000000000
  1.00000000000000
  3.00000000000000
  5.53846153846154 +22.70586296571332i
  0.72675072346706 + 3.48845049166890i
  0.39660995327941 + 5.42673370973554i
 -0.05647681883508 + 7.23146495246913i
 -0.08152497901497 + 9.33778592829862i
  0.00691565645190 + 8.98031608261529i
  0.00007048962414 + 8.99989279646736i
  0.00000000915722 + 8.99999999641514i
 -0.00000000000000 + 9.00000000000000i


iter =

     9

diary off
```