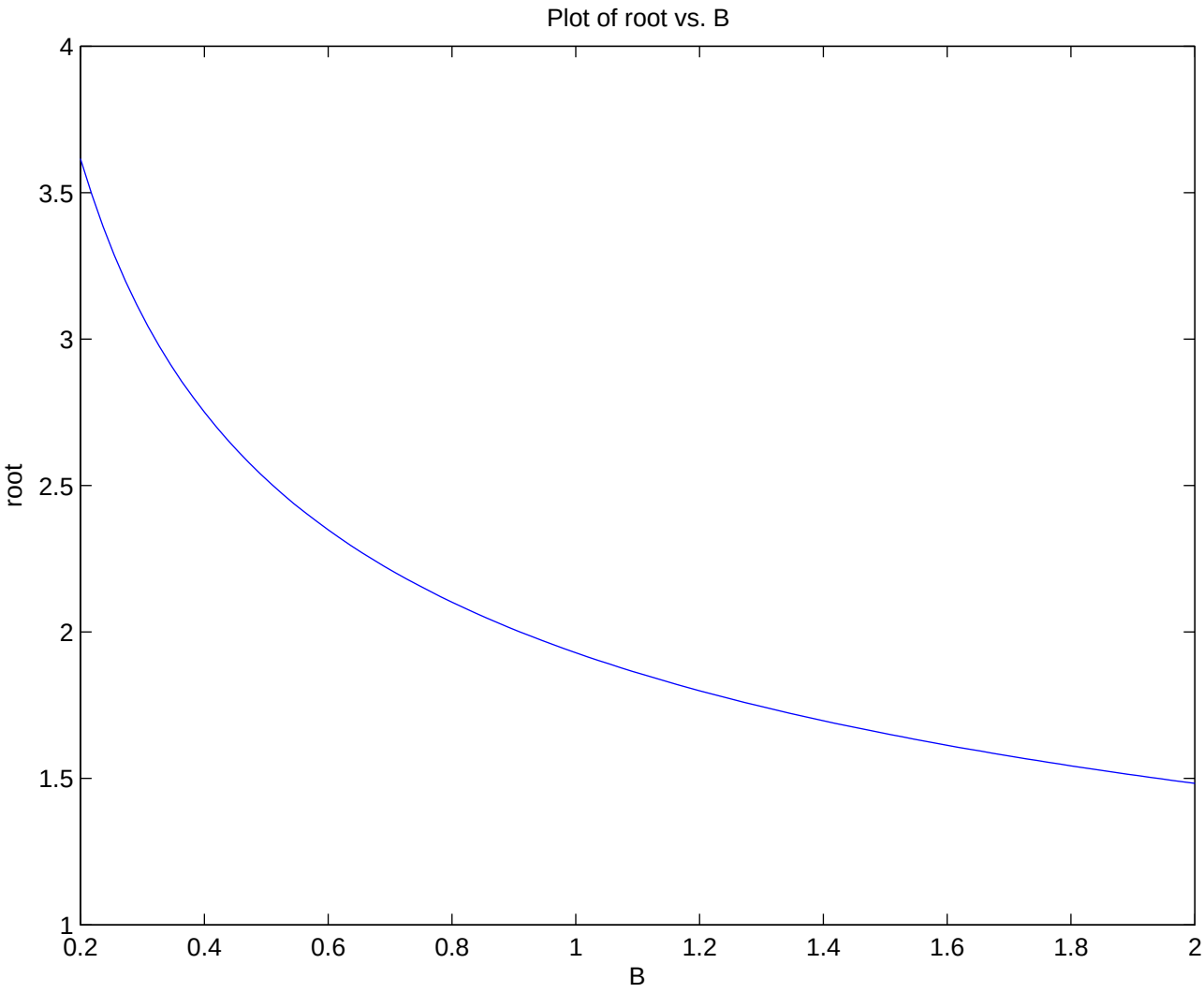
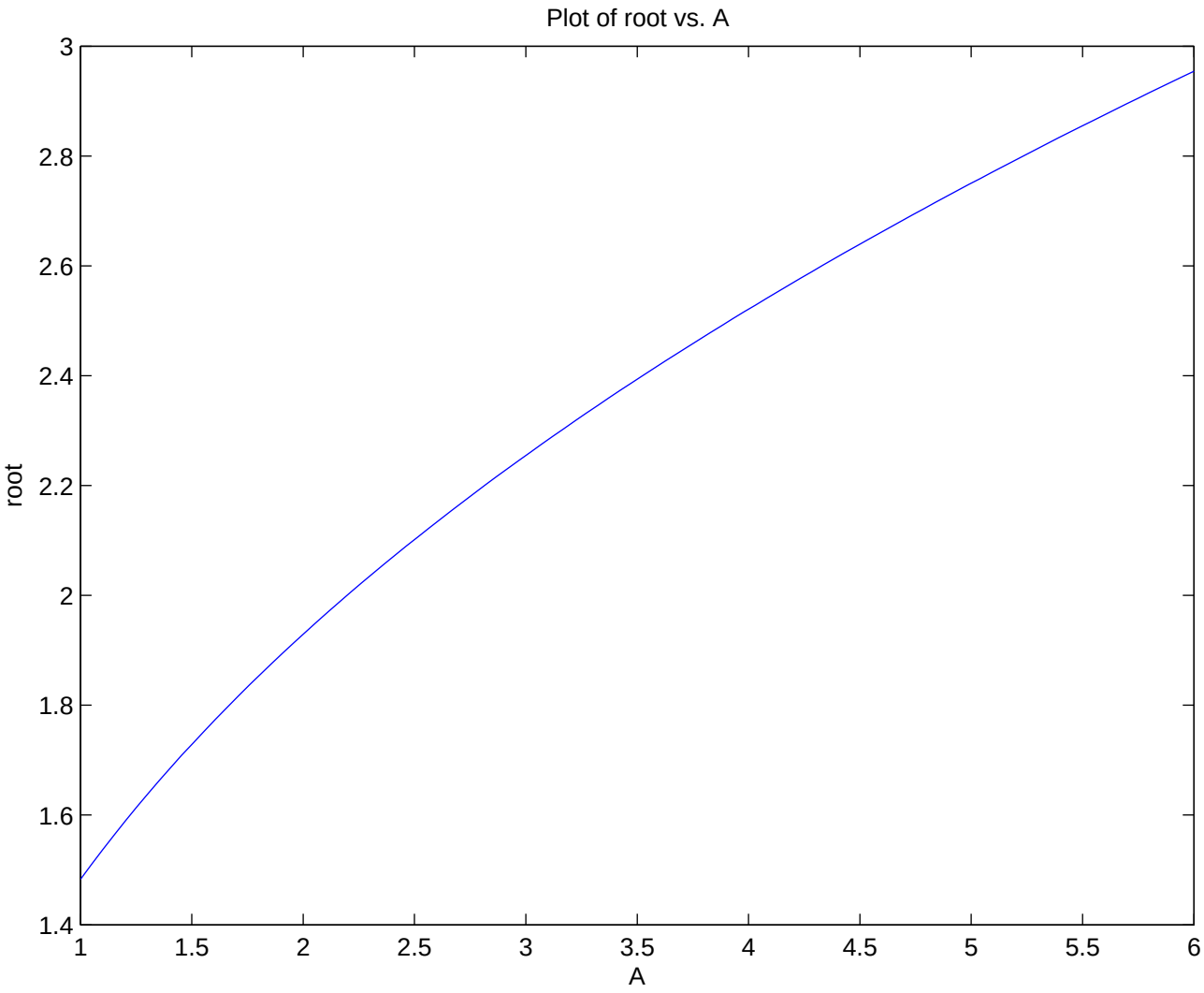
```
function [xnP1, N, xn] = cee451_1(A, B, x0, Nmax, delta)
% problem 1.1. the code.
syms x;
f = A - B.*x.^2.*log(x+1);
dfdx = diff(f, 'x');
xn = x0;
N = 0;

    while(N < Nmax)
        xnP1 = eval(xn - subs(f./dfdx, 'x', xn));
        N = N + 1;
        if  abs(xnP1 - xn) < delta; break; end
        xn = xnP1;
    end


end
```

```
% cee451 p1_2
clc;
clear;
[x, N] = cee451_1(1, 0.2, 0.5, 100, 1e-6);
disp('Problem 1.2');
fprintf('The root %f and the iteration number is %d\n', x, N);
```

```
clear;
A = 4;
B = linspace(0.2, 2, 100);
x = linspace(1, 100, 100);
N = linspace(1, 100, 100);
for i = 1:100
    [x(i), N(i)] = cee451_1(A, B(i), 0.5, 100, 1e-6);
end
disp('2.3(a)')
disp('   x                   B');
disp(horzcat(x',B'));
p1_3a = plot(B,x);
xlabel('B')
ylabel('root')
title('Plot of root vs. B')
saveas(p1_3a, 'p1_3a', 'pdf');

A = linspace(1, 6, 100);
B = 0.5;
x = linspace(1, 100, 100);
N = linspace(1, 100, 100);
for i = 1:100
    [x(i), N(i)] = cee451_1(A(i), B, 0.5, 100, 1e-6);
end
disp('2.3(b)')
disp('   x                   A');
disp(horzcat(x',A'));
p1_3b = plot(A,x);
xlabel('A')
ylabel('root')
title('Plot of root vs. A')
saveas(p1_3b, 'p1_3b', 'pdf');
```


Plot of root vs. B
root
B
4
3.5
3
2.5
2
1.5
1
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2


Plot of root vs. A
root
A
3
2.8
2.6
2.4
2.2
2
1.8
1.6
1.4
1
1.5
2
2.5
3
3.5
4
4.5
5
5.5
6

```
Problem 1.2
The root 2.101702 and the iteration number is 8
>> p1_3
2.3(a)
   x                  B
   3.616051604900626   0.200000000000000
   3.493076102985799   0.218181818181818
   3.383862481860943   0.236363636363636
   3.285972792246125   0.254545454545455
   3.197537989984256   0.272727272727273
   3.117097308081309   0.290909090909091
   3.043489905081853   0.309090909090909
   2.975779778274427   0.327272727272727
   2.913202493055234   0.345454545454545
   2.855126602209238   0.363636363636364
   2.801025189038661   0.381818181818182
   2.750454532889821   0.400000000000000
   2.703037878701224   0.418181818181818
   2.658452925326257   0.436363636363636
   2.616422064284527   0.454545454545455
   2.576704680666015   0.472727272727273
   2.539091019509928   0.490909090909091
   2.503397254242238   0.509090909090909
   2.469461487859869   0.527272727272727
   2.437140484934249   0.545454545454545
   2.406306981382999   0.563636363636364
   2.376847454834023   0.581818181818182
   2.348660265031187   0.600000000000000
   2.321654093694684   0.618181818181818
   2.295746628362690   0.636363636363636
   2.270863446283093   0.654545454545455
   2.246937063323301   0.672727272727273
   2.223906119757963   0.690909090909091
   2.201714680207780   0.709090909090909
   2.180311629247505   0.727272727272727
   2.159650147576960   0.745454545454545
   2.139687256339337   0.763636363636364
   2.120383419330688   0.781818181818182
   2.101702194587726   0.800000000000000
   2.083609928255490   0.818181818181818
   2.066075484790070   0.836363636363636
   2.049070008497000   0.854545454545455
   2.032566712184297   0.872727272727273
   2.016540689352830   0.890909090909091
   2.000968746881191   0.909090909090909
   1.985829255607987   0.927272727272727
   1.971102016587438   0.945454545454546
   1.956768141107634   0.963636363636364
   1.942809942824930   0.981818181818182
   1.929210840591545   1.000000000000000
   1.915955270743052   1.018181818181818
```

1.903028607774031	1.036363636363636
1.890417092467992	1.054545454545455
1.878107766665862	1.072727272727273
1.866088413958751	1.090909090909091
1.854347505678079	1.109090909090909
1.842874151631604	1.127272727272727
1.831658055099157	1.145454545454546
1.820689471658590	1.163636363636364
1.809959171461708	1.181818181818182
1.799458404622920	1.200000000000000
1.789178869420885	1.218181818181818
1.779112683046299	1.236363636363636
1.769252354657809	1.254545454545455
1.759590760533364	1.272727272727273
1.750121121126685	1.290909090909091
1.740836979858208	1.309090909090909
1.731732183487289	1.327272727272727
1.722800863927888	1.345454545454546
1.714037421383662	1.363636363636364
1.705436508690514	1.381818181818182
1.696993016765542	1.400000000000000
1.688702061070922	1.418181818181818
1.680558969009946	1.436363636363636
1.672559268180091	1.454545454545455
1.664698675414950	1.472727272727273
1.656973086553009	1.490909090909091
1.649378566876855	1.509090909090909
1.641911342171375	1.527272727272727
1.634567790354034	1.545454545454546
1.627344433634362	1.563636363636364
1.620237931163474	1.581818181818182
1.613245072137739	1.600000000000000
1.606362769323732	1.618181818181818
1.599588052974317	1.636363636363637
1.592918065108171	1.654545454545455
1.586350054127730	1.672727272727273
1.579881369749551	1.690909090909091
1.573509458232253	1.709090909090909
1.567231857872884	1.727272727272727
1.561046194760629	1.745454545454545
1.554950178767927	1.763636363636364
1.548941599763888	1.781818181818182
1.543018324035547	1.800000000000000
1.537178290903585	1.818181818181818
1.531419509520107	1.836363636363636
1.525740055836999	1.854545454545455
1.520138069734191	1.872727272727273
1.514611752297942	1.890909090909091
1.509159363239935	1.909090909090909
1.503779218448641	1.927272727272727
1.498469687664992	1.945454545454546

```
   1.493229192274960   1.963636363636364
   1.488056203212128   1.981818181818182
   1.482949238963828   2.000000000000000

2.3(b)
   x                   A
   1.482949238963828   1.000000000000000
   1.510724680460184   1.050505050505051
   1.537710658862382   1.101010101010101
   1.563964932539387   1.151515151515152
   1.589538699798354   1.202020202020202
   1.614477586443393   1.252525252525253
   1.638822451111334   1.303030303030303
   1.662610047713196   1.353535353535354
   1.685873574698925   1.404040404040404
   1.708643133826026   1.454545454545455
   1.730946115950514   1.505050505050505
   1.752807527494151   1.555555555555556
   1.774250268329626   1.606060606060606
   1.795295369605504   1.656565656565657
   1.815962198325034   1.707070707070707
   1.836268634167566   1.757575757575758
   1.856231223004242   1.808080808080808
   1.875865310741870   1.858585858585859
   1.895185160479382   1.909090909090909
   1.914204055441892   1.959595959595960
   1.932934389739396   2.010101010101010
   1.951387748658689   2.060606060606061
   1.969574979921506   2.111111111111111
   1.987506257116245   2.161616161616162
   2.005191136324930   2.212121212121212
   2.022638606813531   2.262626262626263
   2.039857136526177   2.313131313131313
   2.056854713017302   2.363636363636363
   2.073638880366596   2.414141414141414
   2.090216772546524   2.464646464646465
   2.106595143648789   2.515151515151515
   2.122780395322360   2.565656565656566
   2.138778601729971   2.616161616161616
   2.154595532290944   2.666666666666667
   2.170236672444792   2.717171717171717
   2.185707242641325   2.767676767676768
   2.201012215738245   2.818181818181818
   2.216156332965868   2.868686868686869
   2.231144118600047   2.919191919191919
   2.245979893468333   2.969696969696970
   2.260667787400348   3.020202020202020
   2.275211750721150   3.070707070707071
   2.289615564875646   3.121212121212121
   2.303882852262361   3.171717171717172
   2.318017085348922   3.222222222222222
```

2.332021595127818	3.272727272727273
2.345899578974899	3.323232323232323
2.359654107957546	3.373737373737374
2.373288133640061	3.424242424242424
2.386804494427554	3.474747474747475
2.400205921485850	3.525252525252525
2.413495044271420	3.575757575757576
2.426674395702165	3.626262626262626
2.439746416997034	3.676767676767677
2.452713462209951	3.727272727272727
2.465577802481226	3.777777777777778
2.478341630027583	3.828282828282828
2.491007061890130	3.878787878787879
2.503576143457884	3.929292929292930
2.516050851783016	3.979797979797980
2.528433098702615	4.030303030303030
2.540724733780524	4.080808080808081
2.552927547081735	4.131313131313132
2.565043271790789	4.181818181818182
2.577073586684733	4.232323232323232
2.589020118470345	4.282828282828283
2.600884443994596	4.333333333333334
2.612668092336612	4.383838383838384
2.624372546788785	4.434343434343434
2.635999246734098	4.484848484848484
2.647549589426207	4.535353535353536
2.659024931678348	4.585858585858587
2.670426591466689	4.636363636363637
2.681755849453333	4.686868686868687
2.693013950433842	4.737373737373737
2.704202104713775	4.787878787878788
2.715321489418422	4.838383838383838
2.726373249739663	4.888888888888889
2.737358500123570	4.939393939393939
2.748278325402149	4.989898989898990
2.759133781872396	5.040404040404041
2.769925898325604	5.090909090909091
2.780655677029712	5.141414141414141
2.791324094667253	5.191919191919192
2.801932103231334	5.242424242424242
2.812480630881909	5.292929292929293
2.822970582764465	5.343434343434343
2.833402841793113	5.393939393939394
2.843778269399949	5.444444444444445
2.854097706252430	5.494949494949495
2.864361972940432	5.545454545454546
2.874571870634510	5.595959595959596
2.884728181716836	5.646464646464646
2.894831670386167	5.696969696969697
2.904883083238151	5.747474747474747
2.914883149822169	5.797979797979798

```
   2.924832583175862   5.848484848484849
   2.934732080338425   5.898989898989899
   2.944582322843687   5.949494949494950
   2.954383977193928   6.000000000000000

>>
```

```
function I = p2(f, x1, x2, N)
step = (x2-x1)/N;
x = x1:step:x2;
I = 0;

for i = 1:N
    I = I + eval(1/2*step*(subs(f,'x',x(i))+subs(f,'x',x(i+1))));
end

end
```

```
function [result, N] = p2r(f, x1, x2, delta, Nmax)
% use p2 function to run
% calculate I for N = 1
N = 1;
I = p2(f, x1, x2, N);
result = 1:1:(Nmax-1);
result = result*0;
result(1) = I;

while N < Nmax
    % calculate I for N = N+1
    N = N + 1;
    I = p2(f, x1, x2, N);
    % store I into a vector
    result(N) = I;
    % test delta
    if abs(2*(result(N)-result(N-1))/(result(N)+result(N-1)))<delta
        break;
    end

end
result = result(1:N);

if N==Nmax
    disp('Attention: NMax is reached!');
end

end
```

```
% main rountine to run
clear;
clc;
syms x;
delta = 1e-3;
x1 = 1;
x2 = 4;
%%
lambda = 1;
f1 = exp(-lambda*x);
[r1 N1] = p2r(f1,x1,x2,delta,100);

%%
f2 = exp(sin(x)+sqrt(x));
[r2 N2] = p2r(f1,x1,x2,delta,100);
%%
fprintf('Problem 2\nmy choice of delta=%f\n',delta);
disp('function 1');
disp('N');
fprintf('%d\n',N1);
disp('I')
fprintf('%3.3f\n',r1);

disp('function 2');
disp('N');
fprintf('%d\n',N2);
disp('I')
fprintf('%3.3f\n',r2);
```

```
Problem 2
my choice of delta=0.001000
function 1
N
12
I
0.579
0.413
0.378
0.366
0.360
0.357
0.355
0.354
0.353
0.352
0.352
0.351
function 2
N
12
I
0.579
0.413
0.378
0.366
0.360
0.357
0.355
0.354
0.353
0.352
0.352
0.351
>>
```