

```

format long;
exit=0;
exitprompt='Want to go again (0=yes/ 1=no)? ';
%{
cc=123456711; %pick a number to find the square root of
act=sqrt(cc); %actual square root
x0=3; %initial approximation$
nTimes=22; %number of times to iterate recursion

for n=1:nTimes
    [a,b]=my_sqrt(cc,x0,n);
    formatSpec="after n=%d iterations we have approx=%.16f and b=%d.";
    sprintf(formatSpec,n,a,b)
end

L1=[];
L2=[];
L3=[];
L4=[];
Lnn=[];
for loop for making lists
for n=1:nTimes
    call the function
    [a,b]=my_sqrt(cc,x0,n);
    c=abs(a-sqrt(cc)); %get difference between apprx and actual
    L1=[L1,a];
    L2=[L2,b];
    L3=[L3,c];
    L4=[L4,act];
    Lnn=[Lnn,n];
end
print out a table
table(Lnn',L4',L1',L2',L3','VariableNames',↵
{'Num_Iterations','Actual','Approx','Number_Agree','Diff'})
%}
while exit ~= 1
    prompt1='I want to find the square root of ';
    x_in=input(prompt1);
    prompt2='My first approximation is ';
    x_0=input(prompt2);
    prompt3='I want the error to be less than ';
    tol=input(prompt3);
    [Sqr_out, err, nIter,nAgree,L1,L2,L3,L4]=my_sqrt(x_in,x_0,tol);
    formatSpec = 'The approximation after %i iterations is %f with error in squares %d.↵
\n The actual square root is %f, our approximation agrees for %i digits';
    sprintf(formatSpec,nIter,Sqr_out,err,sqrt(x_in),nAgree)
    table(L3',L1',L2',L4','VariableNames',↵
{'Iteration','Approx','Diff_in_Squares','Number_Agree'})
    exit=input(exitprompt);
end

```