

```

clc
clear all

%*****
%
% INPUTS
%
% Click the run button and refer to the command window
% These are the inputs that can be modified by the user
%

f= @(x) 2000*log(1400/21./x)-9.8*x ;

% a, the lower limit of integration

a=8 ;

% b, the upper limit of integration

b=10 ;

% n, the maximum number of segments

n=128 ;

%*****

disp(sprintf('\n\nConvergence of the Romberg Method'))
disp(sprintf('University of South Florida'))
disp(sprintf('United States of America'))
disp(sprintf('kaw@eng.usf.edu\n'))

disp(sprintf('\n*****Introduction*****'))

disp('The following simulation illustrates the convergence of Romberg Method')
disp('applied to numerically integrate functions. This section is the')
disp('only section where the user interacts with the program. The user ')
disp('enters a function in the form f(x), the lower and upper limit of integration,')
disp('a and b, and the number of subdivisions to take. By entering this data, the')
disp('program will calculate the exact (Matlab numerical value if it is not exact)')
disp('value of the solution, followed by the results using the Romberg Method')
disp('with 1, 2, 4, 8, 16 ... n segments. The program will also display the true error,')
disp('the absolute relative percentage true error, the approximate error, the absolute')
disp('relative approximate percentage error, and the least number of significant ')
disp('digits correct all as a function of number of segments.')
```

```

disp(sprintf('\n\n*****Input
Data*****\n'))
disp(sprintf('      f(x), function which defines the integrand '))
disp(sprintf('      a = %g, lower limit of integration',a))

```

```
disp(sprintf('      b = %g, upper limit of integration',b))
disp(sprintf('      n = %g, maximum number of segments',n))
format short g

% Exact solution
exact = quad(f,a,b) ;

nstep = floor(log2(n))+1 ;

% First obtain the first approximations via trapezoidal rule
for i=1:nstep

    NN(i)=2^(i-1) ;
    h=(b-a)/NN(i) ;

    integral = 0.5*f(a) + 0.5*f(b) ;
    for j=1:NN(i)-1
        integral = integral + f(a+h*j) ;
    end
    integral = integral * h ;

    RR(1,i)=integral ;

end

% Using the Romberg formula, improve these approximations
index = nstep-1;
for k = [2:nstep]
    for j = [1:index]
        RR(k,j) = RR(k-1,j+1) + (RR(k-1,j+1) - RR(k-1,j))/(4^(k-1)-1);
    end
    index = index - 1;
end

% Calculate errors
for i=1:nstep
    YY(i)=RR(i,1) ;
    Et(i)=exact-integral ;
    Etabs(i)=abs((integral-exact)/exact) ;
    if(i > 1)
        Ea(i)=YY(i)-YY(i-1) ;
        Eaabs(i)=abs((YY(i)-YY(i-1))/YY(i-1)) ;
        SD(i)=floor((2-log10(Eaabs(i)/0.5))) ;
        if(SD(i)<0)
            SD(i)=0
        end
    else
        Ea(1)=0 ;
        Eaabs(1)=0 ;
        SD(1)=0 ;
    end
end
```

```

end

disp(sprintf('\n\n*****Table of Values*****\n'))

disp('      Approx      True      Relative      Approx      Rel Appr      Sig      ')
disp(' n      Integral      Error      True Error      Error      Error      Digits ')
disp('-----')

for i=1:nstep

if(i > 1)
    if(exact || YY(i) > 0)
        disp(sprintf('%4i  %+1.3e  %+1.3e  %+1.3e  %+1.3e  %+1.3e  %2i',NN(i),YY(i),Et(i),
Etabs(i),Ea(i),Eaabs(i),SD(i) ))
    else
        disp(sprintf('%4i  %+1.3e  %+1.3e      n/a      %+1.3e      n/a      n/a',NN(i),YY(i),
Etabs(i),Ea(i)))
    end
else
    disp(sprintf('%4i  %+1.3e  %+1.3e  %+1.3e      n/a      n/a      n/a',NN(i),YY(i),Et
(i),Etabs(i)))
end

end

disp('-----')

% The following generates 3 plots. This function detects information about your
% screensize and tries to then place/size the graphs accordingly.
scnsize = get(0,'ScreenSize');

% Graph 1: Approximation and True Errors
fig2=figure ;
set(fig2,'Position',[0.2*scnsize(3),0.2*scnsize(3),0.6*scnsize(3),0.2*scnsize(4)]) ;
subplot(1,3,1); plot(NN,YY,'-o','LineWidth',2,'Color',[1 0 0]);
title('Appr. Integral vs No. of Segments')

subplot(1,3,2); plot(NN,Et,'-o','LineWidth',2,'Color',[0 0 1]);
title('Et vs No. of Segments')

subplot(1,3,3); plot(NN,Etabs,'-o','LineWidth',2,'Color',[0 0 1]);
title('Abs et vs No. of Segments')

% Graph 2: Relative Errors and Significant Digits
fig = figure ;
set(fig,'Position',[0.2*scnsize(3),0,0.6*scnsize(3),0.2*scnsize(4)]) ;
subplot(1,3,1); plot(NN(2:nstep),Ea(2:nstep),'-o','LineWidth',2,'Color',[0 1 0]);
title('Ea vs No. of Segments')

```

```
subplot(1,3,2); plot(NN(2:nstep),Eaabs(2:nstep),'-o','LineWidth',2,'Color',[0 1 0]);  
title('Abs ea vs No. of Segments')
```

```
subplot(1,3,3); plot(NN(2:nstep),SD(2:nstep),'-o','LineWidth',2,'Color',[1 0.5 0.5]);  
title('Significant Digits Correct vs No. of Segments')
```