```
clc
clf
clear all

%**********************************************************************
%
% INPUTS
%
% Click the run button and refer to the command window
% These are the inputs that can be modified by the user
%
% x,y Data pairs in either ascending or descending order

  x = [-0.5,-0.25,0,0.2,0.5] ;
  y = [0.01,0.005,1,0.5,0.01] ;

%**********************************************************************

disp(sprintf('\n\nSimulation of the Trapezoidal 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('This simulation finds the approximate value of the integral under the curve ')
disp('when the data is given only at discrete data points. Two methods are presented,')
disp('discrete trapezoidal rule and polynomial interpolation.')

disp(sprintf('\n\n*******************************Input↙
Data*********************************\n'))


% Finding the number of data points
n=length(x);
flag_ascend=0;
for i=1:n-1
    if x(i+1)>x(i)
      flag_ascend=flag_ascend+1;
    end
end
flag_descend=0;
for i=1:n-1
    if x(i+1)<x(i)
      flag_descend=flag_descend+1;
    end
end
if flag_ascend~=n-1 & flag_descend~=n-1
    fprintf ('The x values are not in ascending or descending order \n\n')
    stop
end
```

```
% Using the Trapezoidal Rule for discrete data
int_value=0;
for i=1:n-1
   int_value=int_value+(y(i+1)+y(i))/2*(x(i+1)-x(i));
end

fprintf('The integral value using discrete Trapezoidal rule method is = %e \n',int_value)

%
% Using the polynomial fit to intepolate the data to find the integral
% under the curve
% Finding the coefficients of the 'n-1'th order polynomial

p = polyfit(x,y,n-1);
% Using the formula int(x^n)=x^(n+1)/(n+1) on each polynomial part
up=0;
low=0;
for i=1:n
    up = up+p(i)*x(n)^(n-i+1)/(n-i+1);
    low = low+p(i)*x(1)^(n-i+1)/(n-i+1);
end
int_value_poly=up-low;
fprintf('The integral value using polynomial interpolation is = %e \n',int_value_poly)


m=1000;
for i=1:m+1
    xx(i)=(i-1)*(x(n)-x(1))/(m)+x(1);
end
yy=polyval(p,xx);


% Plotting the discrete data showing data points, the straight line
% splines and the polynomial approximation through the data points
plot(x,y,'bo')
hold on;
plot(x,y,'r','LineWidth',1.0)
hold on;
plot(xx,yy,'b','LineWidth',1.0)
title('Plot of y vs x discrete data');
xlabel('\bf\itx');
ylabel('\bf\ity');
legend('Data Point','Straight Spline Approximation','Polynomial Approximation');
grid off;
hold off;
```