% L5Q3: A script to plot a three-point line
% rotated by a user-specified amount
% a user-specified number of times and
% scaled by a user-specified amount
% each rotation.
%
% Usage: L5Q3
%
% Arguments: None.
%
% Returns: Nil.
%
% Author: Lecturers
% Date: Semester 2

theta = input('Enter the rotation angle in degrees: ') * (pi/180);
% Read the desired number of rotations from the user.
rots = input('Enter the number of rotations: ');
% Read the scale factor from the user.
scale = input('Enter the scaling factor: ');

% Initialise three points.
p1 = [1

            2];
p2 = [2

            2];
p3 = [4

            5];
% Construct a matrix of the three points.
The first row of the matrix will contain the x coordinates of
% the points. The second row of the matrix will contain the y
% coordinates of the points.
pts = [p1 p2 p3];
% Construct a rotation matrix for the given theta value.
R = [cos(theta), -sin(theta)

     sin(theta), cos(theta)];
% Loop over the number of desired rotations.
for ii = 0 : rots
    % For each rotation:
    % Plot the y coordinate values versus the x coordinate values.
    % Extract the x coordinate values from the first row of pts
    % matrix.
    % Extract the y coordinate values from the second row of pts
    % matrix.
    line(pts(1, :) , pts(2, :));
    % Multiply the pts matrix by the rotation matrix and scale
    % factor to produce the new point coordinates corresponding
    % to rotation by theta and scaling by scale.
    pts = R * pts * scale;
end
% Force the spacing of the ticks on the axes to be equal.
axis equal;
% Create axis labels and titles.
xlabel('x');
ylabel('y');
title('The Rotated Set Of Points');