function [error\_train, error\_val] = ...

learningCurve(X, y, Xval, yval, lambda)

%LEARNINGCURVE Generates the train and cross validation set errors needed

%to plot a learning curve

% [error\_train, error\_val] = ...

% LEARNINGCURVE(X, y, Xval, yval, lambda) returns the train and

% cross validation set errors for a learning curve. In particular,

% it returns two vectors of the same length - error\_train and

% error\_val. Then, error\_train(i) contains the training error for

% i examples (and similarly for error\_val(i)).

%

% In this function, you will compute the train and test errors for

% dataset sizes from 1 up to m. In practice, when working with larger

% datasets, you might want to do this in larger intervals.

%

% Number of training examples

m = size(X, 1);

[a,n]=size(X);

% You need to return these values correctly

error\_train = zeros(m, 1);

error\_val = zeros(m, 1);

% ====================== YOUR CODE HERE ======================

% Instructions: Fill in this function to return training errors in

% error\_train and the cross validation errors in error\_val.

% i.e., error\_train(i) and

% error\_val(i) should give you the errors

% obtained after training on i examples.

%

% Note: You should evaluate the training error on the first i training

% examples (i.e., X(1:i, :) and y(1:i)).

%

% For the cross-validation error, you should instead evaluate on

% the \_entire\_ cross validation set (Xval and yval).

%

% Note: If you are using your cost function (linearRegCostFunction)

% to compute the training and cross validation error, you should

% call the function with the lambda argument set to 0.

% Do note that you will still need to use lambda when running

% the training to obtain the theta parameters.

%

% Hint: You can loop over the examples with the following:

%

% for i = 1:m

% % Compute train/cross validation errors using training examples

% % X(1:i, :) and y(1:i), storing the result in

% % error\_train(i) and error\_val(i)

% ....

%

% end

%

% ---------------------- Sample Solution ----------------------

for i = 1:m

X1=zeros(i,n);

y1=zeros(i,1);

X1=X(1:i,:)

y1=y(1:i)

% theta=zeros(n,1);

[theta]= trainLinearReg(X1,y1,lambda)

[J,grad]=linearRegCostFunction(X1,y1,theta,0);

error\_train(i)=J;

error\_val(i)=linearRegCostFunction(Xval,yval,theta,0);

% -------------------------------------------------------------

% =========================================================================

end