**MATLAB Code for Hurricane Tracking Model
Using the Distance Formula**

% MATLAB CODE FOR HURRICANE TRACKING MODEL USING THE DISTANCE FORMULA

% Initial Location of the City of Interest may be defined as the Origin (x1,y1)

% where X1 = 0 and Y1 = 0.

% If a different city is selected as the origin, the student may enter any

% values they wish for X1 and Y1.

x1 = 0;

y1 = 0;

speed = 0;

% Student uses the scale of the map to calculate the exact (x1, y1)and(x2, y2)

% location of the hurricane. Note: This program does not use the latitude

% nor longitude values. Part of the lesson is to have students use the

% scale of the map instead to determine horizontal and vertical distances in miles.

%

x1 = input ('Enter the X1 value in miles (X1) = ');

y1 = input ('Enter the Y1 value in miles (Y1) = ');

x2 = input ('Enter the X2 value in miles (X2) = ');

y2 = input ('Enter the Y2 value in miles (Y2) = ');

City = [x1,y1];

hurricanepoint = [x2,y2];

disp ('The initial linear distance in miles of the hurricane to our city is');

% This section of the code uses the Distance Formula to calculate and

% display the linear distance based on the input. This code may be used to

% calculate the distance between any two points.

aPointMatrix = repmat(City,size(hurricanepoint,1),1);

LinearDistance = (sum(((aPointMatrix-hurricanepoint).^2),2)).^0.5

format short g

% The code now prompts the student to enter the travelling speed and time

% interval they wish to calculated.

speed = input ('Enter the travelling speed of the hurricane in mph (SPEED) = ');

time = input ('Enter the time interval in hours (HOURS) = ');

% The code uses simple formulas to calculate and display the initial

% accumulated time, distance, and time to landfall. The xlswrite command

% opens an Excel 2010 spreadsheet.

j=LinearDistance-(speed\*time);

accumtime=0;

nextlocation=LinearDistance;

timeland=LinearDistance/speed;

initialdata = {accumtime LinearDistance timeland};

xlswrite ('hurricane.xlsx',initialdata,'data','A3');

disp ('From this distance, the HURRICANE WILL HIT LAND in exactly');

disp(timeland);

disp ('HOURS');

fid = fopen('hurricane.dat','w');

% The i=1:j is the counter command. It defines how many times the FOR-IF-END

% loop will run. The conditional FOR-IF-END loop calculates the accumulated

% time, distance, and time to landfall for the time interval entered.

for i=1:j;

 newdistance = nextlocation - (speed\*time);

 if newdistance > 0

 accumtime = accumtime + time;

 nextlocation = nextlocation - (speed\*time);

 timeland = timeland -time;

 end

 % The fprintf command inside this LOOP creates the hurricane.dat file.

 fprintf(fid,'%4.8f %4.8f %4.8f\n',accumtime,nextlocation,timeland);

end

% The following commands display various interesting data calculations.

disp ('IF THE HOURS ELAPSED = ');

disp(accumtime);

disp (' ');

disp ('BE PREPARED! THERE ARE ONLY THIS MANY HOURS LEFT BEFORE THE HURRICANE MAKES LANDFALL');

disp (timeland);

disp (' ');

disp ('The data on the hurricane.dat file and Excel Spreadsheet show distances from our city and time to landfall every ');

disp (time);

disp ('HOUR INTVERVAL');

disp ('Enter the word "hurricane" at the prompt to start over');

fclose(fid);

% The following commands load the data from the FOR-IF-END LOOP into the

% hurricane.dat file and the Excel 2010 spreadsheet. The code produces

% hundreds of calculations but once the results start repeating, only the

% initial data points are usable.

hurricane\_data = load('hurricane.dat');

heading1 = {'HURRICANE TRACKING RESULTS'};

xlswrite('hurricane.xlsx',heading1,'data','A1');

heading2 = {'Accumulated Time (hours)', 'Distance to Landfall (miles)', 'Time to Landfall (hours)'};

xlswrite('hurricane.xlsx',heading2,'data','A2');

xlswrite ('hurricane.xlsx',hurricane\_data,'data','A4');

% This command creates the plot that pops up on the screen.

plot(hurricane\_data(:,1),hurricane\_data(:,2),'b-\*');

xlabel('ELAPSED TIME'); ylabel('DISTANCE BEFORE LANDFALL');

%

% THANK YOU FOR USING THIS MATLAB CODE FOR YOUR LESSON.

% From: Armando Vital, math teacher, Veterans HS, Brownsville ISD, TX

% If you have any questions, email me at avital@bisd.us