MATLAB HOMEWORK 4

Jordan Lian

GE 1502 -- 308 Hurtig Hall

10:30 - 11:35am

Professor Whalen

Table of Contents

Land Surveying

Main m-file

```
n_sides = input('Please enter the number of sides of the property --> ');
a = 0;

%initial coordinate
initial_coord(1) = input('Enter initial x coordinate --> ');
initial_coord(2) = input('Enter initial y coordinate --> ');

next_coord = initial_coord;

FID = fopen('property.txt', 'w');
fprintf(FID, 'X Coordinate(ft)\t\t\tY Coordinate(ft)\t\t\tDistance(ft)\t\tBearing (N AZM) -- D\t\t\t   M
\t\t       S');


for i = 1 : n_sides - 1

   %distance
   distance = input('Enter the distance of leg --> ');
   original_angle = input('Enter Bearing in [Degrees Minutes Seconds] --> ');
   bearing = input('Enter 1, 2, 3, or 4 for NW, NE, SE, SW Bearing --> ');

   %bearing test -- to find angle to use with trig
   if (bearing == 1)
      reference_angle = [90, 0, 0] + original_angle;
      disp(reference_angle); %for ensure math was done correctly
      printed_angle = [360, 0, 0] - original_angle;
      if (printed_angle(2) < 0 | printed_angle(3) < 0)
         printed_angle = [359, 59, 60] - original_angle;
      end
   elseif (bearing == 2)
      reference_angle = [90, 0, 0] - original_angle;
      if (reference_angle(2) < 0 | reference_angle(3) < 0)
         reference_angle = [89, 59, 60] - original_angle;
      end
      disp(reference_angle);
      printed_angle = original_angle;
   elseif (bearing == 3)
      reference_angle = [270, 0, 0] + original_angle;
```

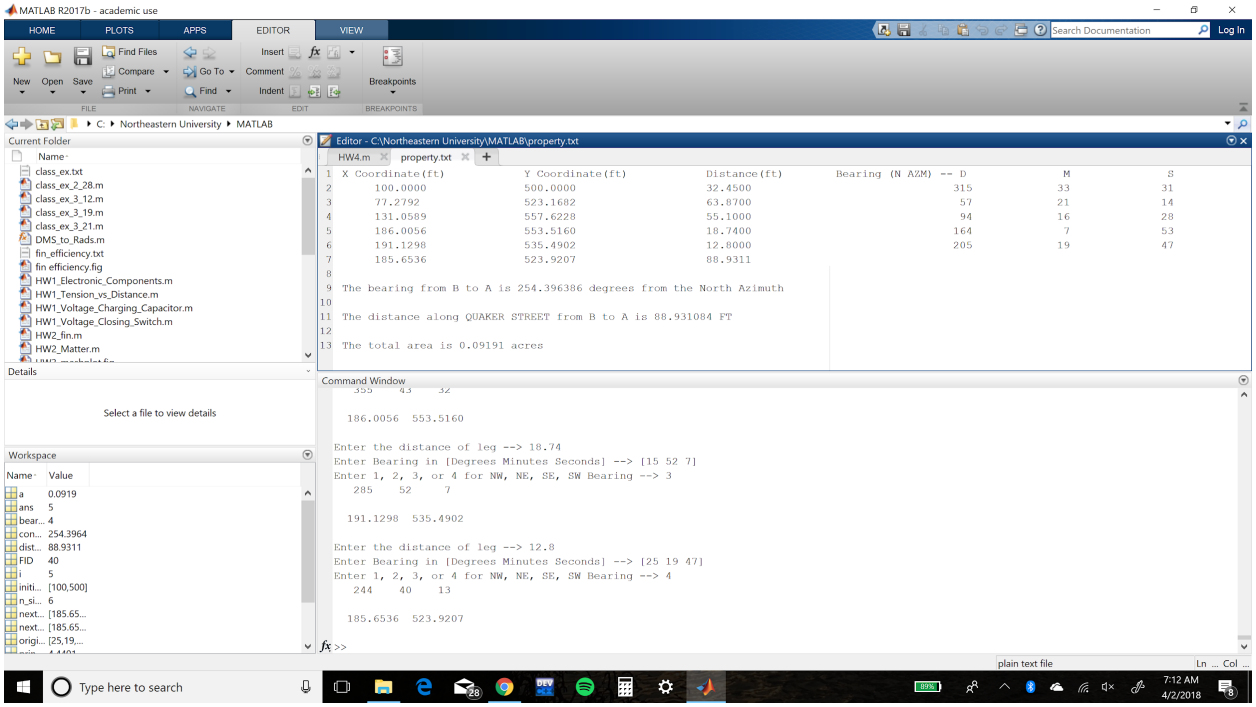
```
        disp(reference_angle);
        printed_angle = [180, 0, 0] - original_angle;
        if (printed_angle(2) < 0 | printed_angle(3) < 0)
            printed_angle = [179, 59, 60] - original_angle;
        end
    elseif (bearing == 4)
        reference_angle = [270, 0, 0] - original_angle;
        if (reference_angle(2) < 0 | reference_angle(3) < 0)
            reference_angle = [269, 59, 60] - original_angle;
        end
        disp(reference_angle);
        printed_angle = [180, 0, 0] + original_angle;
    end

    %convert from DMS to rads
    rads = DMS_to_Rads(reference_angle);

    %get next coordinate
    next_coordinate = Next_Coord(next_coord, distance, rads);
    disp(next_coordinate); %to ensure math and coding were both done correctly

    a = a + (0.5*( (next_coord(1)*next_coordinate(2) ) - (next_coordinate(1)*next_coord(2) ) ) );
%x0y1 - x1yo + x1y2 - x2y1
    %print output
    fprintf(FID, '\n\t %4.4f\t\t\t\t\t%4.4f\t', next_coord);
    fprintf(FID, '\t\t\t\t%4.4f\t\t\t\t', distance);
    fprintf(FID, '\t\t\t\t %4.0f\t\t\t%4.0f\t \t\t%4.0f', printed_angle);

    next_coord = next_coordinate;

end

a = a - (0.5 * ((next_coordinate(2)*initial_coord(1) ) - (next_coordinate(1)*initial_coord(2) ) ) );
%x6yo, y6x0
a = a * -2.29568 * 10^-5; %flip counterclockwise negative area, and ft^2 to acres

%distance
distance = sqrt( (next_coord(1) - initial_coord(1)).^2 + (next_coord(2) - initial_coord(2)).^2);

%angle
ref_angle = (pi/2) - atan((next_coord(2) - initial_coord(2)) / (next_coord(1) - initial_coord(1)));
print_angle = pi + ref_angle;
conv_angle_deg = print_angle * 180 / pi;
```

```
fprintf(FID, '\n\t %4.4f\t\t\t\t%4.4f\t', next_coord);
fprintf(FID, '\t\t\t\t%4.4f\t\t\t\t', distance);

%print remaining results
fprintf(FID, '\n\nThe bearing from B to A is ');
fprintf(FID, '%4.6f ', conv_angle_deg);
fprintf(FID, 'degrees from the North Azimuth\n\n');

fprintf(FID, 'The distance along QUAKER STREET from B to A is ');
fprintf(FID, '%4.6f ', distance);
fprintf(FID, 'FT\n');

fprintf(FID, '\nThe total area is ');
fprintf(FID, '%4.5f ', a);
fprintf(FID, 'acres');
```

Function m-files

```
function rads = DMS_to_Rads(reference_angle)
%degrees to radians function
rads = (reference_angle(1) * pi / 180) + (reference_angle(2) * pi / 180 / 60) +
(reference_angle(3) * pi / 180 / 3600);
end

function [next_coordinate] = Next_Coord(next_coord, distance, rads)
%determining next coordinate
%   need distance, angle, and reference angle
   x_next = next_coord(1) + (cos(rads) * distance);
   y_next = next_coord(2) + (sin(rads) * distance);
   next_coordinate = [x_next, y_next];
end
```

Output file results



Business-Style Letter

(see attached file)