Optics 211 Exam Spring 2024

Instructions:

This exam is due on Blackboard by 11:59:59PM, Sunday, April 21. Late submissions will not be tolerated without written permission of the instructor. You may not request an extension on the exam. Extensions are for emergency circumstances or academic accommodations only.

You must comment all codes you generate and upload all m-files and relevant figure files with your exam submission. This exam is provided to you as a Microsoft Word document. Please use Microsoft Word to complete this exam. Copy output and answers directly into this document (using the proper fonts and formatting as described in the labs). You MUST upload your submission as a Microsoft Word document as I will be commenting on your exam within the document and sending it back to you.

WARNING: Completing this exam in Google Docs WILL result in missing information when uploaded to Blackboard. You MUST complete this in Microsoft Word to avoid missing points due to an incomplete submission.

You must work independently on this exam and not use an AI tools, but otherwise you may use any other resources at your disposal. You must cite any sources you used to solve any problem within your lab submission. Please review instructions on Blackboard for further clarification. You must complete Appendix 1 verifying you worked independently and used no AI tools. Failure to sign this document will result in a zero score for this exam.

All questions about this exam should be directed to Greg or Glory. Any course TA can answer questions about the labs and the course material in general. Please send Greg or Glory a Slack message, get on Zoom during any designated office hours for help, or send an email to Greg or Glory if need be. If you need help, ask. Don't struggle needlessly.

Make sure you follow all problem instructions or you will not receive full credit for the problems. Properly label all figures. Adhere to good presentation standards for all figures. Include scripts, output and command window execution for all problems in this exam.

Upload all m-files generated and relevant figure files with your exam submission. This exam will be curved if a curve is warranted.

Best of luck!
Greg

1) **Short Answer (35 *points*):** *Follow all instructions provided. Use the equation tool in Microsoft Word to create the equations when relevant or use a stylus to very neatly and clearly write out the equations. Copy all command window work right here, into your exam document. If an m-file is used, copy the code of the m-file here. Partial credit may be awarded for this problem.*

A. Create the following three independent variable arrays using a single command for each array in the Matlab command window and without doing any calculations to create any of the arrays (either on paper or in Matlab):

```
%question A

% i

w = linspace(-2*pi,2*pi,8)p1

%ii

yx = [0 1 2 3 4 5 6 7 8 9 10]

%iii

g3 = 16:.5:20
```

i. A variable named w consisting of 8 equally spaced points inclusive of the lower bound = -2π and the upper bound = 2π.

ii. A variable named y(x) inclusive of the range from 0 to 10 in whole number steps.

iii. A variable named g_3 with a range of 16 to 20 and a spacing of 0.5 between points in the array.

For each part of 1.A, you should copy in your command window command execution showing the both the command and its output. Leave out any semicolons so the variables are displayed within the command window. Represent *y(x)* and g_3 in a way consistent with correct Matlab syntax considerations. *Note: We learned of different ways to make arrays in Matlab. Review this material. You should use a different method for creating the array for each of the three parts of this problem.*

B. Plot the following two decaying exponential functions over the range given on a single set of axes. Your plot must be fully labelled and formatted so that it is presentation quality.

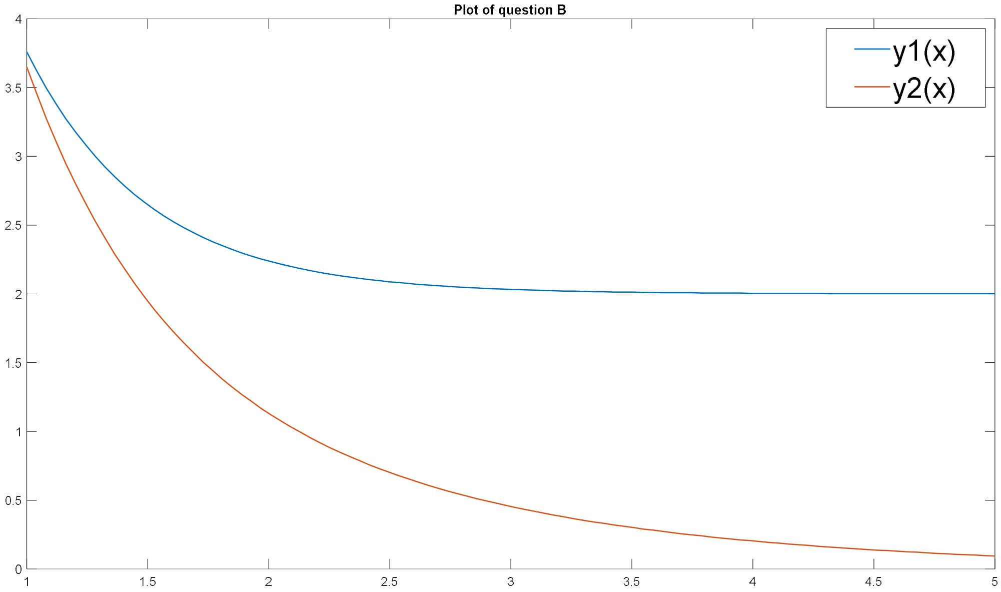
$$y_1(x) = 13e^{-2x} + 2$$

$$y_2(x) = 13e^{-2x} + 4e^{-0.75x} + 2$$

$$0 \le x \le 5$$

You must complete this problem entirely within the command window and edit your plot to make it presentation quality entirely within the figure window. Be sure your plot includes the origin in the bottom left corner of the plot.

```
%question B

x = linspace(1,5);

y1x = 13.*exp(-2.*x)+2;

y2x = 13.*exp(-2.*x)+4*exp(-0.75.*x);

plot(x,y1x,x,y2x)
%if run each line at a time (I.E copy and paste each line separately and run them one
after another)
%in the command window it will still function
%the labeled plot will be attached as a .fig file
```



C. Complete the following matrix operations in the Matlab command window:

i. Define the following matrix:

$$\boldsymbol{A} = \begin{bmatrix} \mathbf{-5} & \mathbf{1} & \mathbf{2} \\ \mathbf{4} & \mathbf{6} & \mathbf{-3} \\ \mathbf{1} & \mathbf{-2} & \mathbf{1} \end{bmatrix}$$

ii. Is matrix A potentially invertible? If yes, compute the inverse of *A* using Matlab.

iii. Multiply matrix A by the 3x3 identity matrix using the built-in identify matrix function in Matlab. You should not define the identity matrix manually. Does the mathematical result of this problem make sense given what the identity matrix does?

iv. In a single command, select the second column of matrix *A* and save that column as a new 3x1 matrix, *B*.

v. In a single command and without redefining the entire matrix B, change just the value of the second row of *B* from 6 to a new value of -5. This matrix should still be saved as matrix *B*.

vi. Combine *A* and *B* to create matrix *C* – a new 3x4 matrix of the form:

$$\boldsymbol{C} = [\boldsymbol{A}\ \boldsymbol{B}]$$

You must accomplish with a single command and without redefining either matrix.

vii. Use the `rref` function to determine the form of any solutions to the three-variable linear system represented by matrix *C*. Show the form of the solutions here (If there are any. If there are not, simply state that there are no solutions).

```
% Question C

% %i
% A = [-5 1 2;4 6 -3;1 -2 1];
%
% %ii
% inv(A);
%
% %iii
%
% eye(3)*A;
%
% %iv
%
% B = A(:,2);
%
% %v
%
```

```
   % B(2,1) = -5;
   %
   % %vi
   %
   %  C = [A B];
   %
   %  %vii
   %
   %  rref(C);
   %
   %  %the answers to the system are -1.57, -1.2, -2.83
   %if run each line at a time (I.E copy and paste each line separately and run them
one after another)
   %in the command window it will still function
```

D. Write a simple function as an m-file. This function should accept a single numerical input from the user. The function should use conditional statements to discriminate between the following three cases:

1. The input number is negative.
2. The input number is precisely zero.
3. The input number is positive.

For each case, output a written string which tells the user whether the number is negative, positive, or zero depending on the case. Please add an `else` statement which contains an error statement telling the user that the input must be a number. Include a help file for this function. The help file should contain all of the normal expected things and tell the user how to use the script.

Copy your script here and upload the m-file as an attachment to this exam assignment.

```
% function OPT211_Final_EAK
%
% OPT211 Part 1 D 4/20/24
% Is my number positive or negative?
% Inputs include and real valued number such as 1, 352, -76, .049, -0.12
% Outputs include "The input number is positive","The input number is negative","The
input number is precisely zero."
% By Ezra A-K


% user input variable
N = input('Please input a real number between negative infinity and infinity ');
```

```
%checks if the user input is positive or negative
if N > 0
    disp('The input number is positive.');
elseif N < 0
        disp('The input number is negative.');
elseif N == 0
        disp('The input number is precisely zero.');
else
    % i, "For when you have an over-active imagination!"
    error('The input must be a REAL number.');
end
```

Test your script using the command window and inputs which will activate all three cases and the error statement. Copy all four of these command window command executions here:

E. Why is it important to be mindful of how a function of two independent variables is represented in a projection of the function in a two-dimensional plane?

```
% Question E

disp('It is important to recognize what color gradient you are using to represent the
functions and perspective you chose')
disp('because with an improper view or color scheme it can be hard to tell what a
graph is supposed to represent.')
```

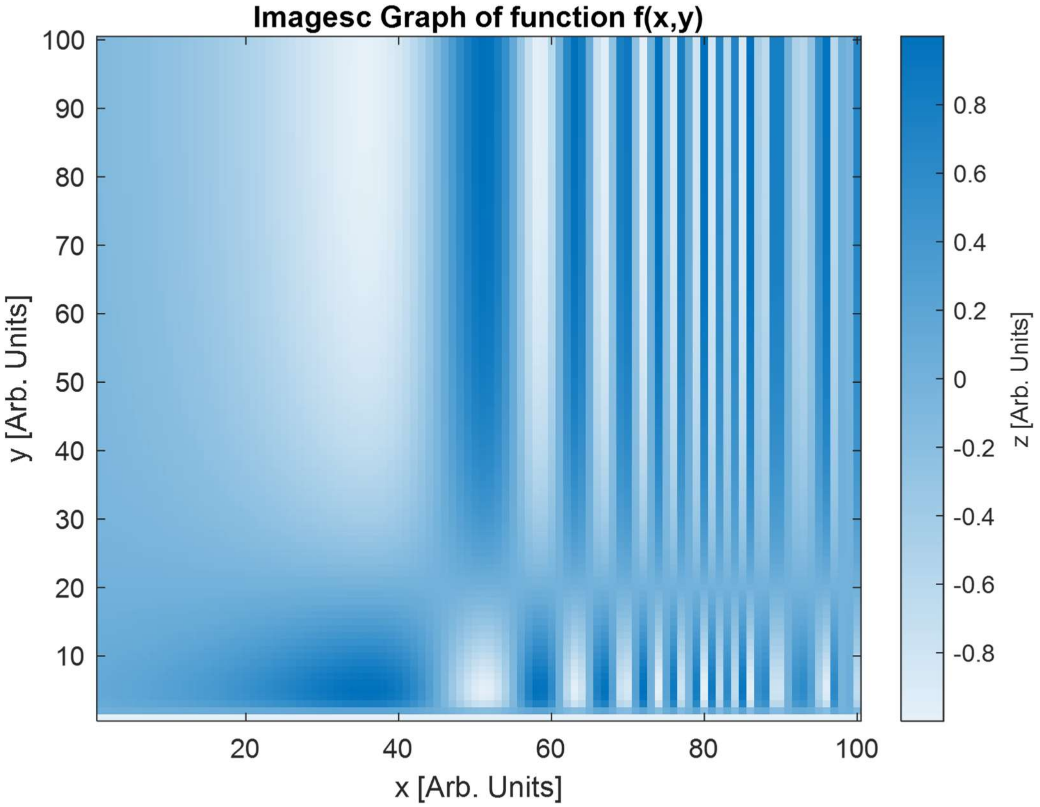
2) 2D Projections of Two-Variable Functions (*25 points*):

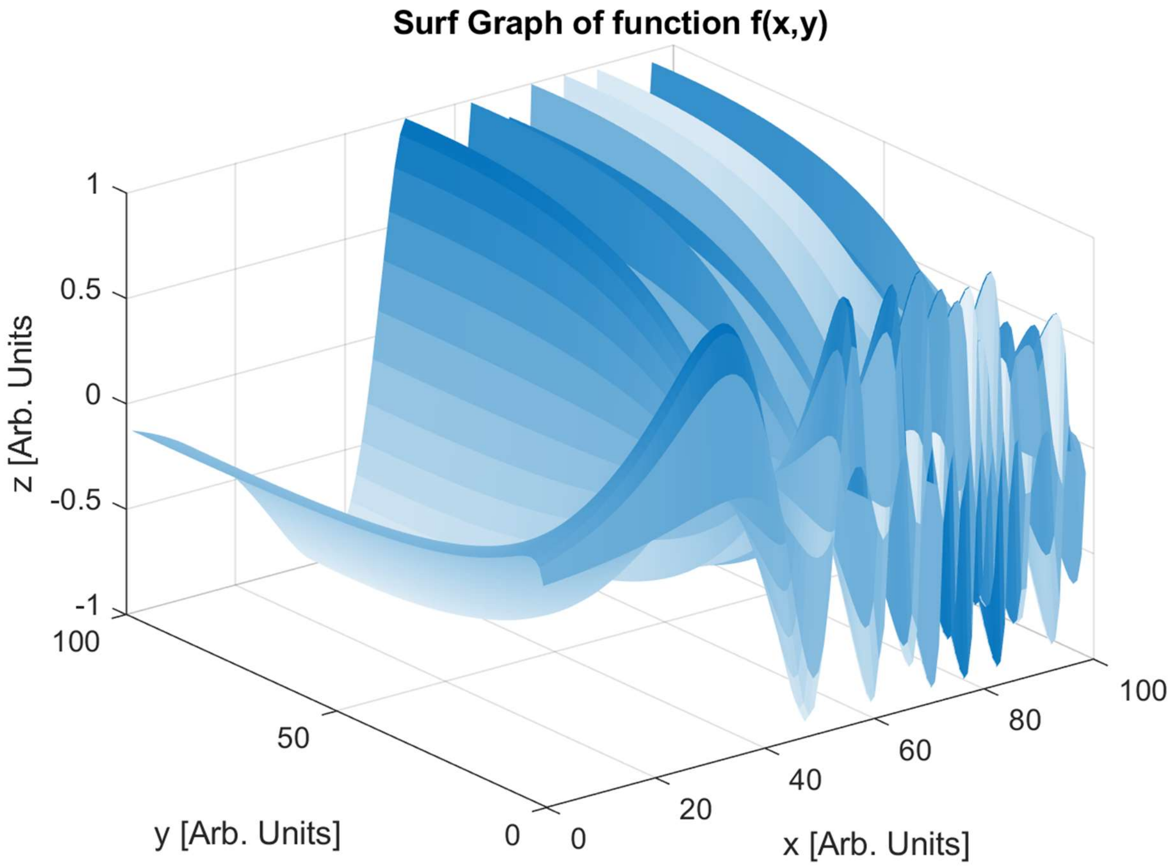
A. Plot the following function of two-variables over the ranges provided.

$$\boldsymbol{f(x,y) = sin(e^{x})cos\big(ln(y)\big)}$$
$$\boldsymbol{-2 \le x \le 5}$$
$$\boldsymbol{0 \le y \le 25}$$

Create two plots: One using the `surf` function and one using the `imagesc` function. Make both projections fully labelled and presentation quality, including labelled colorbars, etc.

Which of the two projections do you think most fairly and accurately displays the function for the user? I think the Imagesc graph is easer to understand since the surf graph is very cluttered.



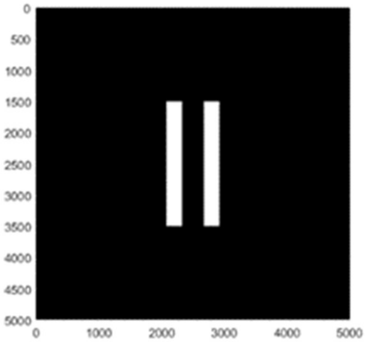


Write an m-file to create the two projections. Copy the script of the m-file here and upload the m-file with your exam submission.

```
%Question A

% creates x and y varibles and thier bounds
x = linspace(-2,5);
y = linspace(0,25);

% combines x and y into one matix for use in function
[x1,y1] = meshgrid(x,y);

%a function of x, y and z
z = sin(exp(x1)).*cos(log(y1));

% creates a surface on a 3D graph of z
figure(1)
surf(z,'MeshStyle','none')
xlabel('x [Arb. Units]')
```

```
ylabel('y [Arb. Units]')
zlabel('z [Arb. Units')
colormap('sky')
title('Surf Graph of function f(x,y)')

% creates a 2D color coded graph of z
figure(2)
imagesc(z)
xlabel('x [Arb. Units]')
ylabel('y [Arb. Units]')
c2 = colorbar;
c2.Label.String = 'z [Arb. Units]';
set(gca,'Ydir','normal') %Get current axis and change direction of y-axis
thanks lab 5
colormap('sky')
title('Imagesc Graph of function f(x,y)')
```

B. Use the information provided in Part 5 of Lab 5 to convert your square aperture to the double slit aperture:



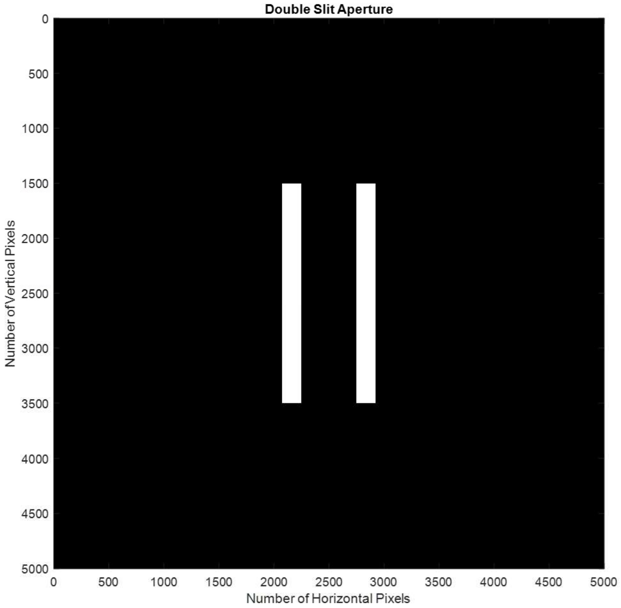
Slit width of 175 pixels (for each slit)

Slit height of 2500 pixels (for each slit)

Slit spacing of 500 pixels (slit center-to-center)

Make the double slit aperture in its own m-file. Copy the script of your m-file here and upload your m-file with your exam submission.

Copy your aperture figure here. Title it, label the x and y axes, and list the units as pixels on each axis. You do NOT need to include a color bar in your figure.

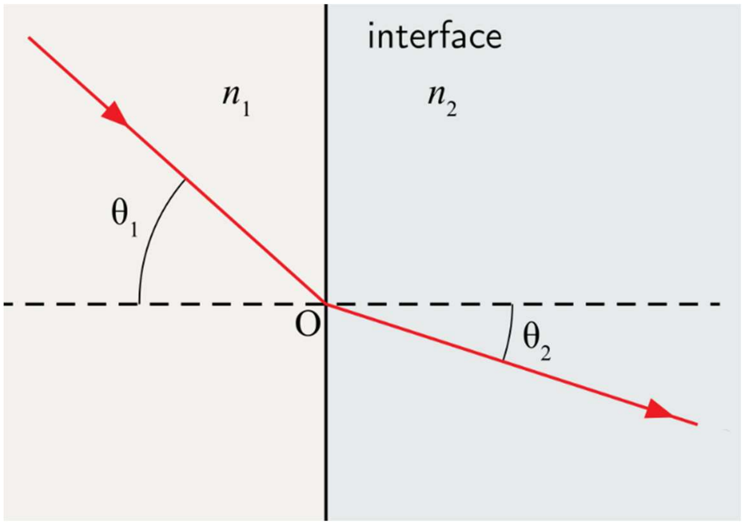


```
% Question B

% Define Aperture field

apl = 5000; % Size of the aperture field
ap = zeros(apl); % Define actual aperture plane
ap(1500:3500 , 2075:2250) = 1; % Defines the left slit
ap(1500:3500 , 2750:2925) = 1; % Defines the right slit

% Plot the aperture field
figure(1)
imagesc(ap) % Plot image of the aperture field
colormap gray % Set the color of the aperture field plot
axis equal % Set the display scale of the axes
axis([0 apl 0 apl]) % Set axes limits to size of aperture field
```

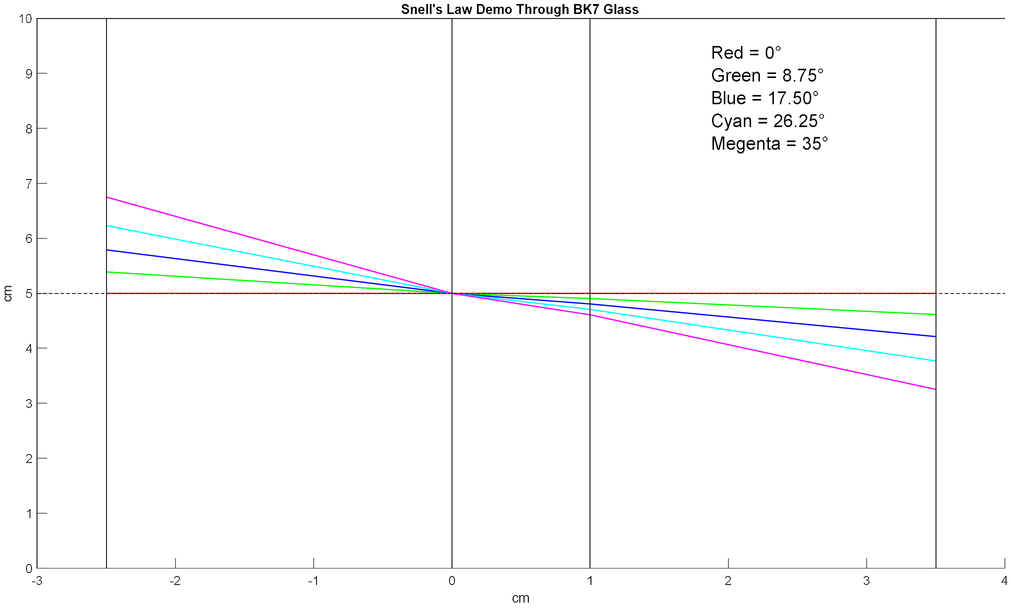
3) Viewing Snell's Law Through 3 Materials *(40 points)*:

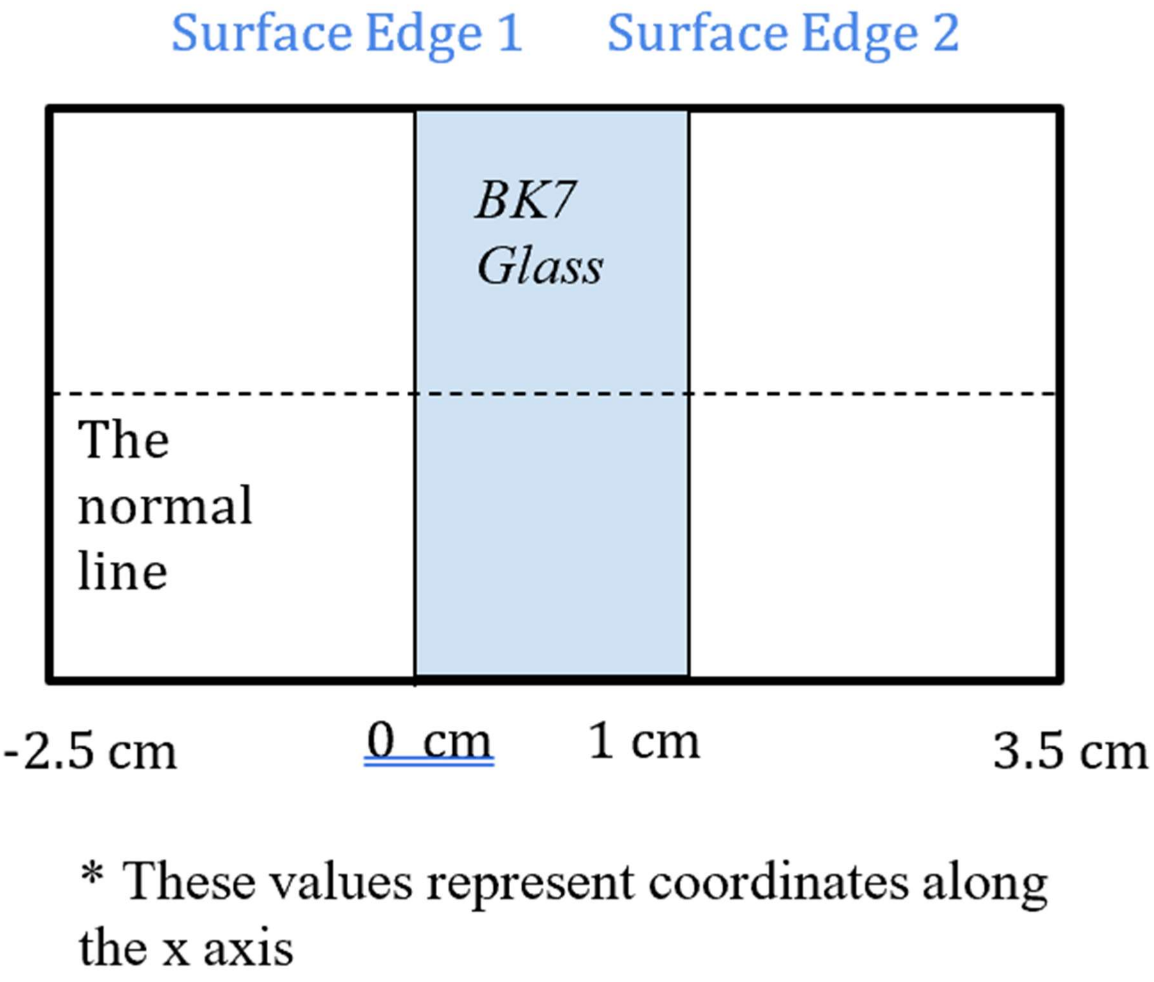
In this problem, you will be tracing 5 different rays of light going from air and through BK7 Schott glass. When ray tracing into different media, Snell's law can be used to determine at what angle and distance the light will exit at each surface. For reference, Snell's law is:

$$n_1 * sin(\theta_1) = n_2 * sin(\theta_2)$$



Where the 1 indicates the first material index and incoming angle at a material interface and the 2 indicates the second material index and angle coming out of the surface edge. For this problem, you will need to use at least one for loop to create a figure that shows 5 different paths of light going through 3 surfaces (their indices labeled below). Your figure should be formatted like below with an appropriate legend and titles. (labels of the surface edges and normal line are not required).





Critical Information:

- n_1 = 1.0 (index of air)
- n_2 = 1.57 (index of a BK7 glass)
- n_3 = 1.0 (index of air)
- 5 angles equally spaced from 0° to 35°

Requirements:

- Use at least one for loop
- The first edge between the first 2 surfaces must be at 0 cm and must be shown on the plot
- The second edge between the last 2 surfaces must be at 1 cm and must be shown on the plot

- The 5 angles from 0° to 35° must be evenly spaced, with the first angle being 0° and the final angle being 35°. (The other 3 angles must be spaced evenly between those points)
- For each angle, the light pathway through the material must be all the same color
- All the angles in the first medium must meet the surface edge at (0,0)

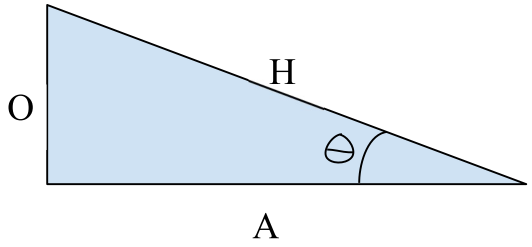
```
% array of angles
ang = linspace(0,35,5);

% will be used to chose angles
theta = 0;

% colors for later
co = ['r', 'g', 'b', 'c', 'm'];

% goes though all angles and calculates starting and end points for rays
% I have shifted everything up by 5 because I don't like the axis not being
% at 0 at the corner and also it's clearer to see the normal line
figure(1)
for  j = 1:5
    theta = ang(1,j);

    % outside of glass
        x = [-2.5 0];
        y = [5+2.5*tand(theta) 5];
        line(x,y,'color', co(j),'lineWidth',1)

    % inside of the glass
        x = [0 1];
        y = [5 5-tand(asind((sind(theta))/1.57))];
         line(x,y,'color', co(j),'lineWidth',1)

    % outside of the glass
        x = [1 3.5];
        y = [5-tand(asind((sind(theta))/1.57)) 5-2.5*tand(theta)];
         line(x,y,'color', co(j),'lineWidth',1)

    % The x's define the starting and ending points of the x cordinates of
    % the line. The y's define the starting and ending points of the y
    % cordinates of the line.


% makes the defining lines
yline(5,'LineStyle','--')
yline(0)
yline(10)
xline(-2.5)
xline(3.5)
xline(0)
xline(1)
end
```

```
% title
title("Snell's Law Demo Through BK7 Glass")

% axis labels
xlabel('cm')
ylabel('cm')

%CREATETEXTBOX(figure1)
%  FIGURE1:  annotation figure

%  Auto-generated by MATLAB on 21-Apr-2024 16:28:46

% Create textbox
annotation('textbox',...
    [0.6650625 0.67234219269103 0.150145833333333 0.225409413304698],...
    'String',{'Red = 0°','Green = 8.75°','Blue = 17.50°','Cyan = 26.25°','Megenta = 
35°'},...
    'FontSize',15,...
    'FitBoxToText','off',...
    'EdgeColor',[1 1 1]);
```

Helpful Hints:

- You can make an array of strings! So, for example, if you write down different colors, you can call them up at each iteration of a for loop.
    - Ex. `str = ['Var1'; 'Var2'; 'Var3'; 'Var4'; 'Var4'];`
- You can make vertical and horizontal lines in MATLAB! Check out the help files on the commands, `yline()` and `xline()`
- You can use your array of incident angles as your for loop indices.
  Ex. Instead of saying: `for i = 1:5`

  You can use `for theta` (assuming theta is the array of angles you created)
- When plotting these rays, triangles will be your best friend! Also, SOH-CAH-TOA:



$$\boldsymbol{sin(\theta) = \frac{O}{H};\ cos(\theta) = \frac{A}{H};\ tan(\theta) = \frac{O}{A};}$$

- You can plot multiple points with one plot line! If you take advantage of using an array format, you can plot multiple points: (red is one point, blue is another). This is not needed to complete this problem but it may be helpful.
  <u>Example:</u> `plot([0 1],[2 3],"LineWidth",2)`

Appendix 1:

Academic Honesty Pledge

I, Ezra Alcon-Kirshman, certify that I completed the exam for OPT211: Matlab for Optics Majors I independently and that I did not seek any uncited outside help or help from any other student or any AI tool while completing the exam. My work is my own, and I understand that if it is discovered that I received outside help on the exam, I will receive zero credit for the exam and be reported to the Academic Honesty Board.

I further recognize that cheating is a professional liability which hurts not only myself but hurts others I cooperatively cheat with or anyone I steal work from. I understand cheating is potentially career ending, and I am above it. I will strive to work with integrity throughout my career.

I also recognize the importance of informing relevant parties if I observe cheating. I recognize that although some may view such action in a negative light, it is the right thing to do and is a service to all involved – particularly those observed to be cheating.

Date: 21-Apr-2024 19:03:04

Signature: E.A.K

Name (Printed): Ezra Alcon-Kirshman