

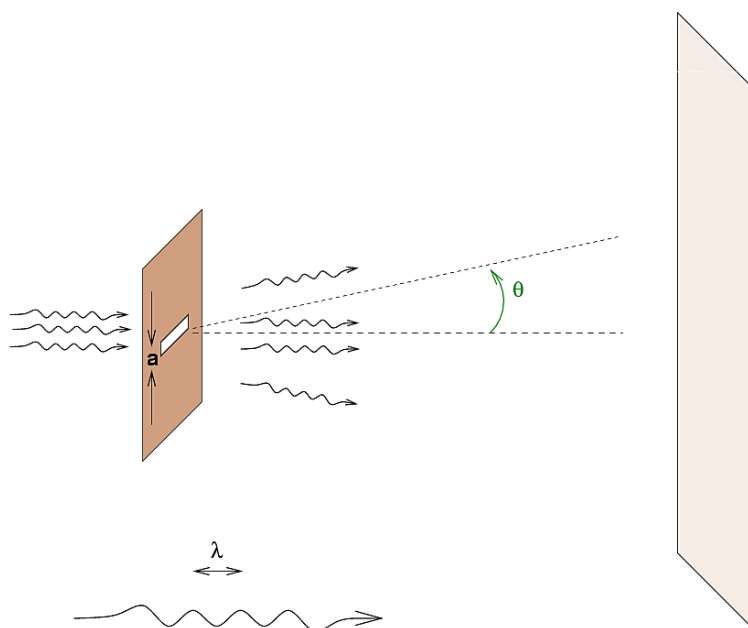
# Math 1B Take-Home Problem for Chapter 7

Read the background information below on the diffraction of light and then answer the following questions.

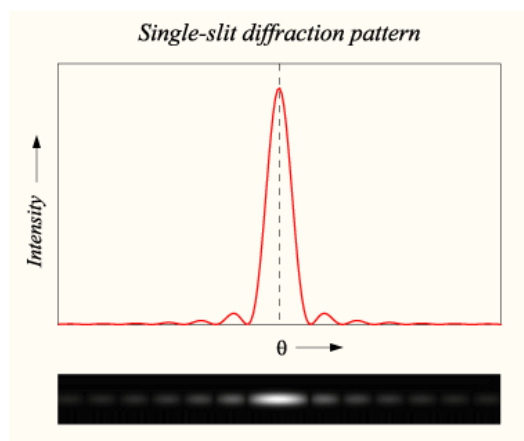
## A brief look at diffraction of light

Light can be considered as a collection of waves. The light waves we can see have wavelengths that range between about 400 and 800 nm. Several very interesting (and useful!) properties of light are a consequence of the way that waves combine with each other; the technical term is **interference phenomena**.

One of these phenomena occurs when light passes through a small opening. A tight bundle of parallel rays entering the opening will spread out after it leaves.



If the opening is in the form a thin slit, then the light rays which pass through it will form a pattern of bright and dark spots, like this:



The pattern is brightest at the center. We can describe this pattern as an intensity (brightness) as a function of the angle away from the central position. This function can be written in the following manner. First, we define an auxiliary variable in terms of the angle **theta**

$$\alpha \equiv \frac{\pi a}{\lambda} \sin(\theta)$$

and then we use it to compute the intensity at the angle **theta** like so:

$$I(\theta) = I_0 \left( \frac{\sin(\alpha)}{\alpha} \right)^2$$

Your job is to integrate this function over various ranges of angle  $\theta$ .

Your assignment is to develop computer code on the platform of your choice (TI8X or Maple or Mathematica, etc) to compute approximations for the following:

- (1) `right(a, b, epsilon)`
- (2) `trap(a, b, epsilon)`
- (3) `midpt(a, b, epsilon)`
- (4) `simp(a, b, epsilon)`

for approximating  $\int_a^b f(x) dx$  using (1) right endpoints as sample points (`right`), (2) the trapezoidal method (`trap`), (3) midpoints as sample points (`midpt`) and (4) Simpson's method (`simp`). The parameter `epsilon` : is the error tolerance for each method, that is, the maximum difference between the true value of the integral and the tolerance.

For example, on the TI85 we can program a "subroutine" called "RSUM" with the TI Basic code below. Note that this program assumes that A, B, N and y1 are already set.

```
:(B-A)/N→H // Compute the length of a subinterval
:0→S // Initialize value of the Riemann sum to zero
:A→x // Initialize variable of integration to left endpt
:For(K,1,N,1) // Set up a for-loop to compute the Riemann sum
: x+H→x // Increment x to right endpoint of next interval
: S+y1→S // Increment the sum of function values
:End // End the for-loop
:S*H→S // Multiply the sum of function values by delta x
```

Now the TI-Basic program below will prompt the user for a function and the bounds of integration and then use the program above to compute a succession of approximating sums until the difference between two successive approximations is less than a user-supplied tolerance.

```

:Fix 5 //Use 5 digits to display a number
:Disp "Enter f to integrate:" //Prompt user for function
:InpSt F // Input the function as a string
:StEq(F,y1) // Convert the string to an equation in y1
:Disp "Enter bounds:" //Prompt user for bounds
:Prompt A,B
:Disp "Enter tolerance:" //Prompt user for tolerance
:Prompt ε
:0→oS //Initialize old Sum to zero
:ClLCD //Clear the screen
:4→N //Initialize number of subintervals parameter
:RSUM //Approximate the integral by right endpoints sum
:While abs (S-oS)>ε //Loop while error is greater than tolerance
: 2*N→N //Double the number of subintervals
: S→oS //Save old sum before computing the new, improved sum
: RSUM //Compute new approximation after doubling N
: I+1→I //This is for displaying output
: Outpt(I,1,N) //Show N in first column
: Outpt(I,5,S) //Show S in second column
: Outpt(I,1,abs(oS-S)) //Show error in third column
:End // End the while loop

```

Here is the output of this program in two screen captures, for the integral  $\int_0^{0.3} (\sin(\sin \theta))^2 d\theta$  first the user prompts and then the output:

```

RIGHT      8.00.01034 .00178
Enter f to integrate: 16.0.00950 8.41978E-4
?(sin(sin x))^2 32.0.00909 4.09288E-4
Enter bounds: 64.0.00889 2.01720E-4
A=?0 128.0.00879 1.00129E-4
B=?3 256.0.00874 4.98816E-5
Enter tolerance:
ε=?1E-4

```

Of course, if you're not up to heavy programming, you might try to use an applet such as this one: <http://www.slu.edu/classes/maymk/Riemann/Riemann.html> .

## Solutions to the questions about the numerical algorithms:

1. Set the wavelength  $\lambda = 1.0$  and the width of the slit  $a = 10.0$ . Use each method (rectangular, midpoint, trapezoidal, and Simpson's) to integrate the intensity of the diffraction pattern from an angle  $\theta = 0$  to  $\theta = 0.3$ . For each of the three methods, you must divide the interval into 16, 32, 64, 128, 256 little pieces. Make a table which shows for each method

# of slices	value of integral for <b>right endpoints</b>	fractional change since previous iteration
16	0.039022194870616	0.009381131350733
32	0.043710767685715	0.004688572815098
64	0.046054521233863	0.002343753548149
128	0.047226263517624	0.001171742283761
256	0.047812100968241	0.000585837450617
512	0.048105011266603	0.000292910298363
1024	0.048251464308793	0.000146453042189
2048	0.048324690303124	0.000073225994331
4096	0.048361303168597	0.000036612865473

# of slices	value of integral for <b>midpoints</b>	fractional change since previous iteration
16	0.048399340500913	0.000003985720539
32	0.048398274782111	0.000001065718802
64	0.048398005801483	0.000000268980628
128	0.048397938418957	0.000000067382527
256	0.048397921565065	0.000000016853891
512	0.048397917351081	0.000000004213984

# of slices	value of integral for <b>trapezoidal method</b>	fractional change since previous iteration
16	0.048395054718673	0.000008271502770
32	0.048397197609788	0.000002142891116
64	0.048397736195945	0.000000538586157
128	0.048397870998710	0.000000134802765
256	0.048397904708829	0.000000033710119
512	0.048397913136943	0.000000008428114
1024	0.048397915244007	0.000000002107065

# of slices	value of integral for <b>simpson's method</b>	fractional change since previous iteration
16	0.048397911906837	0.000000100020564
32	0.048397915724674	0.000000003817837
64	0.048397915932974	0.000000000208301
128	0.048397915945545	0.000000000012570
256	0.048397915946324	0.000000000000779

Using these data as a basis for estimating the error in approximation, we have the follow:

N	Right endpts error	Trapezoid error	Midpoint error	Simpson error
2	0.041560	-0.0334226	0.0334549	-0.0106847
4	0.037509	0.0000176	0.0000047	0.0124112
6	0.025013	0.0000187	-0.0000085	0.0005644
8	0.018757	0.0000112	-0.0000054	0.0000090
9	0.016672	0.0000089	-0.0000043	n/a
8192	0.000018	0.0000000	-0.0000000	0.0000000
15000	0.000010	0.0000000	-0.0000000	0.0000000
16000	0.000009	0.0000000	-0.0000000	0.0000000

2. Using your tables as evidence, answer the following question:

- o Which method converges most rapidly?  
SOLN: As expected, Simpson is fastest, by far, but surprisingly, only once  $n > 16$  or so.
- o How fast does each method converge? That is, if you double the number of slices, does the fractional change go down by a factor of 2? Or a factor of 4? Or what?  
SOLN:  
For right endpoints, the differences reduce by about  $\frac{1}{2}$  when the number of slices doubles.  
For midpt and traps, the differences reduce by about  $\frac{1}{4}$  when the number of slices doubles.  
For simpson's rule, the differences reduce by about  $\frac{1}{20}$  when the slices double, but this is with sampling twice as often as the others.

3. For each method, find out how many pieces into which you must break the interval in order to reach a fractional change of  $1e-5$ . Make a table which shows the number the pieces, and the time it took your program to reach this point and evaluate the integral, for each method. That is, include the time it takes your program to iterate until it reaches the number of pieces required to satisfy the criterion. Which is fastest?

SOLN:

Right endpoints,  $n$  is between 15000 and 16000

Trapezoidal,  $n = 9$

Midpoint rule,  $n = 4$

Simpson's rule,  $n = 8$

Strangely, midpoint nails this threshold before Simpson. Lucky!

Let's now ask a few questions about the physics -- properties of the pattern of light itself. You may use whichever method you wish to answer them. Continue to use  $\lambda = 1$  and  $a = 10$ .

4. At what angle does the first dark spot occur? That is, at what angle will the light intensity first drop to zero?

$$\text{SOLN } I(\theta) = \left( \frac{\sin(10\pi \sin \theta)}{10\pi \sin \theta} \right)^2 = 0 \Leftrightarrow \sin(10\pi \sin \theta) = 0 \Leftrightarrow 10\pi \sin \theta = \pi \Rightarrow$$

$$\theta = \arcsin\left(\frac{1}{10}\right) \approx 0.100167:$$

At what angle does the second dark spot occur?

$$\text{SOLN: } 10\pi \sin \theta = 2\pi \Leftrightarrow \theta = \arcsin\left(\frac{1}{5}\right) \approx 0.201358$$

5. At what angle does the third dark spot occur?

$$\text{SOLN: } 10\pi \sin \theta = 3\pi \Leftrightarrow \theta = \arcsin\left(\frac{3}{10}\right) \approx 0.304693$$

6. What fraction of the total light falls in the central bright spot?

SOLN: Since the light distribution is symmetric about  $\theta = 0$ , we can compute

$$\int_0^{0.100167} \left( \frac{\sin(10\pi \sin \theta)}{10\pi \sin \theta} \right)^2 d\theta \approx 0.0451665 \text{ and}$$

$$\int_0^1 \left( \frac{\sin(10\pi \sin \theta)}{10\pi \sin \theta} \right)^2 d\theta \approx 0.0496641 \text{ and take the portion of light that falls in the central}$$

bright spot as approximately  $\frac{0.0451665}{0.0496641} \approx 90.94\%$

7. What fraction of the total light falls in the second bright spot?

$$\text{SOLN: } \int_{0.100167}^{0.201358} \left( \frac{\sin(10\pi \sin \theta)}{10\pi \sin \theta} \right)^2 d\theta \approx 0.00238157$$

so approximately  $\frac{0.00238157}{0.0496641} \approx 4.80\%$  of the light lies in the two second bright spots.

8. What fraction of the total light falls in the third bright spot?

$$\text{SOLN: } \int_{0.201358}^{0.304693} \left( \frac{\sin(10\pi \sin \theta)}{10\pi \sin \theta} \right)^2 d\theta \approx 0.00085015$$

so approximately  $\frac{0.00085015}{0.0496641} \approx 1.71\%$  of the light lies in the two third bright spots.