Simulation documentation

# Intro

The actual simulation is handled by using the object JScreen.

JScreen uses 3 other objects. It uses “FresnelBiprismX”, “FresnelBiprismX\_Monochromatic” and “SpectrumX”.

JScreen extends JComponent and this object handles the drawing of the simulation. It is supposed to be used by the final GUI program as a Component and handles dynamic resizing.

SpectrumX extends JFrame. As parameters it uses a FresnelBiprismX . Spectrum X basically handles the drawing of values contained and computed by the FresnelBiprismX. Finally it draws all of the resulting paramenters which is a final color, the spectrum, a ruler and a function of the amplitude of each wavelength. This object is called each time the JScreen is clicked at a specific spot form the mouse.

FresnelBiprismX is the object responsible for calculating the final color and spectrum at a distance X from the center of the screen.

FresnelBiprismX\_Monochromatic is an extension of FresnelBiprismX. It does the same thing as the super class but takes into account the source wavelength. This is done by calculating a Gaussian function that deduces which wavelength initially exist, and their respective amplitude.

Wavelength is used by FresnelBiprismX in order to calculate RBG values for each wavelength. This object is a library originally written by Roedy Green, Canadian Mind Products in 1998. The current version used is 1.3 2007-11-27.

To better explain everything we are going to start with the more basic components.

# FresnelBiprismX

This object does the calculations of the light interference at a specific screen distance x in mm.

It has various constructors. The biggest constructor uses 7 arguments, 6 of them double, 1 int.

This constructor gives freedom to change all possible parameters, meanwhile smaller constructors use default values for certain arguments.

The big constructor is like this:

public FresnelBiprismX (double X, double n, double d, double DD, double LOWER\_Range, double UPPER\_Range, int width)

With each of the arguments being:

* (double) X = distance from the center of the screen in mm. [default value is 7]
* (double) n = refraction index of the Biprism. [default value is 1.537]
* (double) d = distance from the light source to the Biprism in cm. [default value is 1]
* (double) DD = distance from the Biprism edge to the screen in cm. [default value is 3]
* (double) LOWER\_Range = the lowest wavelength used for the spectrum in μm. [default value is 0.38]
* (double) UPPER\_Range = the highest wavelength used for the spectrum in μm. [default value is 0.78]
* (int) width = the width of the spectrum i.e. how many wavelengths are used in total to calculate the final color. (if width=3 we only have 3 wavelengths: 380nm,580nm,780nm). The higher this number, the better our final color. [default value is 5000]

The object uses all the arguments in order to specify the initial parameters.

Afterwards it calculates all constants.

It initializes the spectrum which is an array filled with wavelengths and their respective intensities, it calculates the intensity of each wavelength in the spectrum.

And finally all the RGB colors of the spectrum are averaged to find the final color.

For more information go to the object code of the file “FresnelBiprismX.java” where all the code is present with the explanations.

# FresnelBiprismX\_Monochromatic

This object extends FresnelBiprismX and adds one more argument – the variance of the wavelength.

This object has a higher complexity, that’s why it was decided to be a different object.

It could be integrated into FresnelBiprismX, but it was easier to think with two objects in mind, and implementation in the final JScreen was easier as well.

The only method which is different is the function() method which calculates the wavelengths and amplitudes.

For more information go to the object code of the file “FresnelBiprismX\_Monochromatic.java” where all the code is present with explanations.

# SpectrumX

SpectrumX extends JFrame and takes as argument a FresnelBiprismX. And it paints all of its information. So it paints the spectrum and final color with a bench in order to understand the wavelengths.

Below you have examples of SpectrumX.

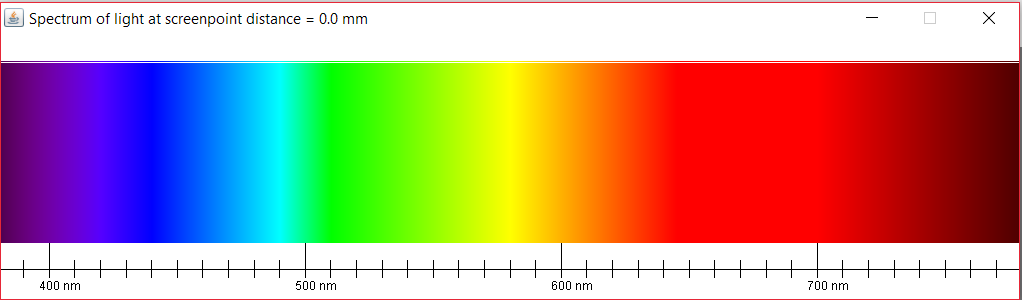


Figure 1 SpectrumX at X=0mm, notice the full spectrum, the final color white at the top and the ruler presenting the wavelengths at the bottom

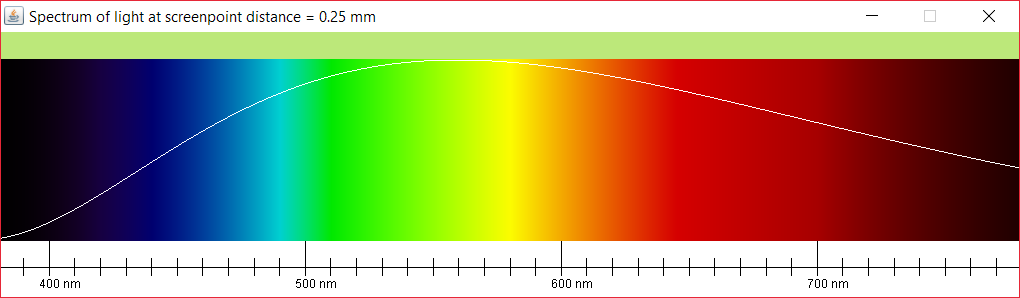


Figure 2 SpectrumX at X=0.25, notice the final color is green-yellowish. The plotted fuction of the amplitude can also be seen

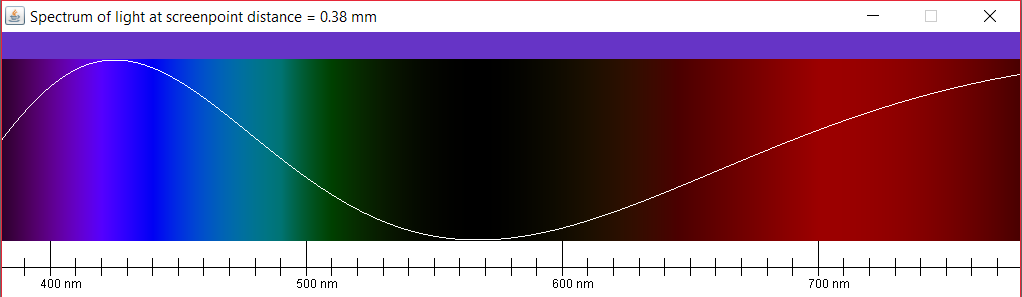


Figure 3 SpectrumX at X=0.38, the final color is purpleish, notice the function as well as the intensities of the colors. They are proportional

For more information go to the object code of the file “SpectrumX.java” where all the code is present with explanations.

## Wavelength

Wavelength is used by FresnelBiprismX in order to decompose specific wavelengths into RGB colors.

This code author info:

@author Roedy Green, Canadian Mind Products

\* @version 1.3 2007-11-27 correct slight error in speed of light.

\* @noinspection WeakerAccess

\* @since 1998

For more information go to the object code of the file “Wavelength.java” where all the code is present.

# JScreen

JScreen extends JComponent. It is meant to be the integral component in the simulation which corresponds to what is shown on the screen in reality during the experiment.

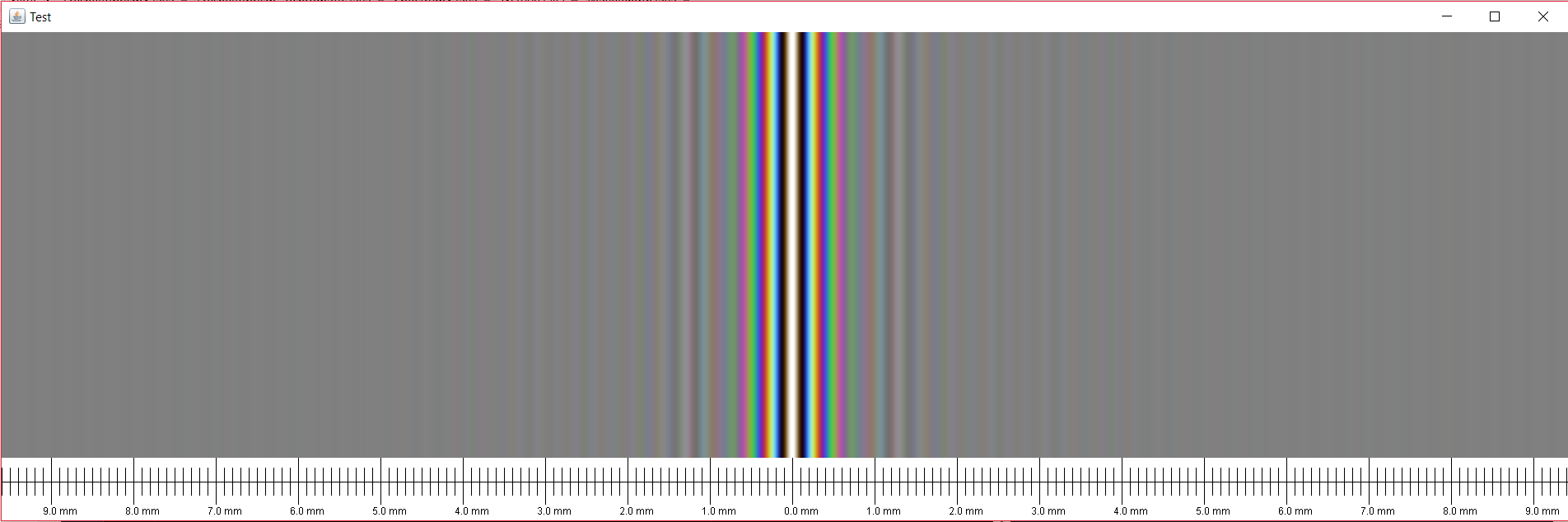
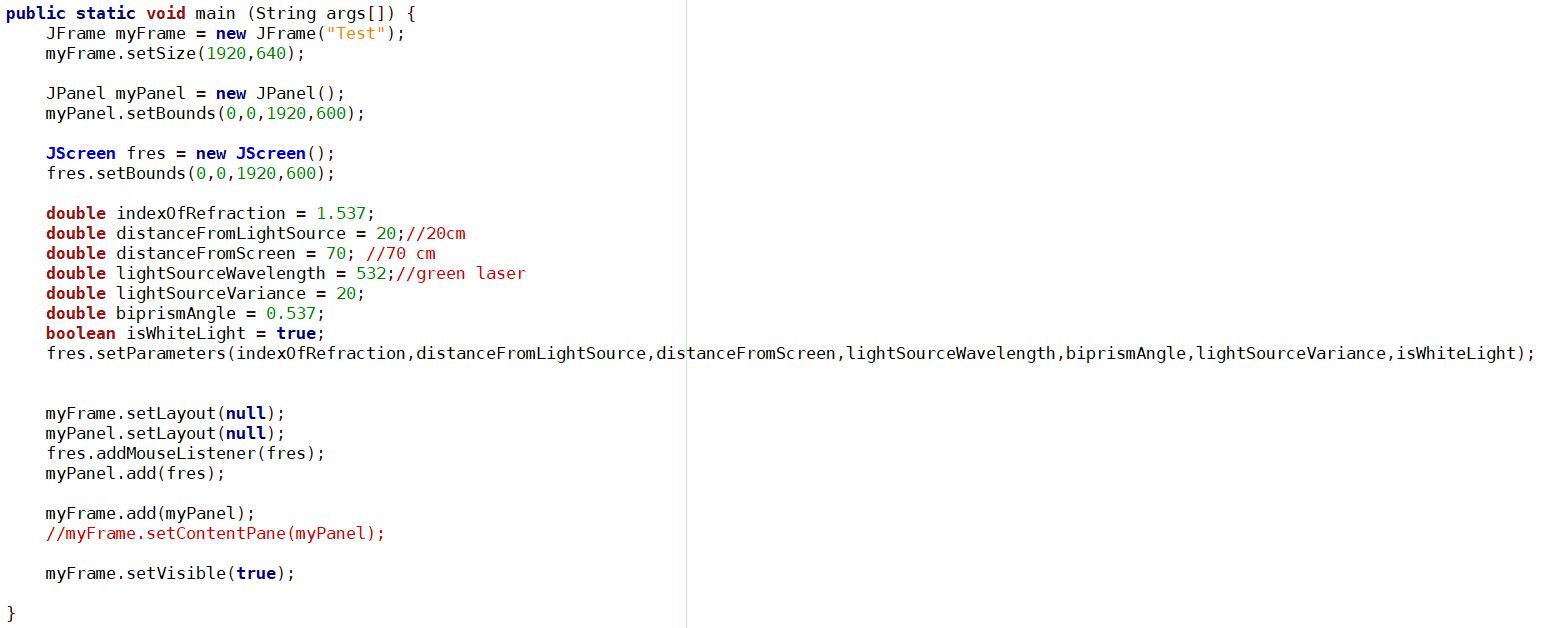


Figure 4 The execution of the default main in the JScreen code. The color spectrum simulation is painted and the screendistance is written below as well. The main creates a JFrame “Test” containing the JScreen with standard parameters.

How to implement the JScreen.

A simple way to implement JScreen can be done by seeing how it works in the main() i.e.:



If you want more details, you can go through the source code and read the comments.