This programming assignment is designed to familiarize you with using the OpenGL API to set up a 3D wireframe animation. To assist you in this assignment, the source code for a complete example program, “Vortex” is linked to the course website (http://www.cs.siue.edu/wwhite/CS482/Syllabus.htm).

Specifically, your program will respond to the user hitting numerical keys by applying Gerstner waveforms to the grid. Gerstner waves are used to demonstrate the effect of the displacement of a surface when a sinusoidal wave passes through. The displacement equations are as follows:

\[
\begin{align*}
    x(t) &= x_0 - \frac{k_x}{|\vec{K}|} A \sin(k_x x_0 + k_z z_0 - \omega t + \varphi) \\
    y(t) &= y_0 + A \cos(k_x x_0 + k_z z_0 - \omega t + \varphi) \\
    z(t) &= z_0 - \frac{k_z}{|\vec{K}|} A \sin(k_x x_0 + k_z z_0 - \omega t + \varphi)
\end{align*}
\]

Where \((x_0, y_0, z_0)\) is the original grid point position, \(A\) is the amplitude of the wave, \(\omega\) is the frequency of the wave, \(\varphi\) is the phase of the wave, and \(\vec{K} = (k_x, k_z)\) is the 2D wave vector. For water waves, \(\omega^2 = g |\vec{K}|\) where \(g\) is the acceleration due to gravity, 9.81 m/sec\(^2\). For instance, with a waveform moving from the grids near left corner to its far right corner, we would use \(k_x = (\frac{1}{2}\sqrt{2})\omega^2\), \(k_z = -(\frac{1}{2}\sqrt{2})\omega^2\), and \(|\vec{K}| = \omega^2\).

In your program, you will enable the user to activate up to four Gerstner waveforms, using keyboard activation, with ‘1’ activating a left-right waveform, ‘2’ activating the aforementioned near-left/far-right waveform, ‘3’ activating a near-far waveform, and ‘4’ activating a near-right/far-left waveform, as illustrated above. Each key will toggle the designated waveform, so having multiple waveforms activated at once will merely produce the sum of the designated waveforms, as illustrated on back of this page. The amplitudes and frequencies of each activated waveform should be randomly generated (amplitudes between 0.2 and 0.7, frequencies between 1.0 and 4.0).
Note that combining multiple Gerstner waveforms in this fashion produces an effect similar to the animation of an ocean surface.

Keep your code modular and readable, with an extensive explanation (including your name) at the top of each program file, explanatory sentences preceding each function, and in-line comments every place within the code where your logic is particularly complicated. You may modify the provided code to formulate your program, but rewrite all comments to reflect what your program is doing. Avoid code redundancy by foregoing cut-and-paste in favor of placing any code that is needed repeatedly into its own module (function, class, structure, etc.).

This is an individual programming assignment, so any examination or use of code other than that which is provided by the instructor or with the course textbooks is expressly forbidden. Properly annotate all code that you utilize from the instructor or the textbooks. Start early on the assignment, and consult with the instructor as soon as possible if questions arise or if problems persist.

Place all of your program files (not your entire project folder) into a single folder named with your last name. When the instructor creates a new Visual C++ project and properly attaches the OpenGL libraries and your code, it must compile and execute in order to be graded. Zip-compress this folder and copy it to your drop-box by Tuesday, November 1, 2016, at 12:00 Noon. Late assignments are not accepted without verifiable medical documentation.