LabVIEW HW #2

Each problem begins with a suggested descriptive name (including the *.vi* extension) for the solution VI that you will write. Suggested icons for use in the VI are given at the end of some problem statements. The palette locations of the cited icons are not give explicitly: these icons can be found with the aid of **Quick Drop**.

1. Complete the Do-It-Yourself/"USE IT!" section ending the chapter on The While Loop and Waveform Chart.

2. **CPU Hogging** So as to explore the impact of While Loop execution on your program's usage of your computer's central processing unit (CPU), …
   a. **For the Windows OS:** Open the "Windows Task Manager" by right-clicking an empty area on the task-bar at the bottom of your monitor and selecting **Start Task Manager** or by pressing <Ctrl+Shift+Esc>.
   b. **For the Mac OS:** Launch the "Activity Monitor" found in your Utilities folder. (Command+Shift+U will take you to that folder. Alternatively, from the Finder, click on the **Go** menu and select **Utilities**.)

   You are now monitoring CPU usage (expressed as a percentage of your computer's maximum value).

   a) Open **Sine Wave Chart [While Loop]** and run this VI with **Wait (ms)** programmed to produce one While Loop iteration every 100 ms. What is the approximate increase in CPU usage that results when the program is running?
   b) Next, stop the VI and re-program **Wait (ms)** to produce one While Loop iteration every 1 ms, and then run the VI. Now what is the approximate increase in CPU usage that results when the program is running?
   c) Finally, stop the VI and delete **Wait (ms)** from the VI's block diagram, so that the While Loop will iterate as fast as possible on your computing system (*e.g.*, on the order of a *micro*second per iteration), and then run the program. What is the approximate increase in CPU usage that results when the program is running, now? [Note Well! Such "CPU hogging" should be avoided unless your program is executing a task where high speed is essential (*i.e.*, not simply updating a plot, as in this case).]