**LabVIEW 101 – Weeks 5**

Before class:

1. Download updates from <https://github.com/rizett/LabVIEW-101>

Required equipment:

Computer with LV installed

Topics:

Exercises

Arrays & Clusters

Reading from files

**Exercises and Timer example**

Open and test / modify / play with week5\_timer.vi (Github/examples and functions/examples). Explore the VI noting its functionality. Adjust the Time to wait value and note what happens. This VI mimics what may occur when running automatic calibrations, for example.

Exercise 1: Open week4\_exercise0 (from the Library).

Add a **Wait** or **Wait until next MS Multiple** function within the while loop to set the frequency that data are plotted. Use a **front panel control** to set the frequency. Pay attention to the time units on the timing function.

Save-as “labview101\_week4\_exercise1” into the LabVIEW101\_library.

Exercise 2: From labview101\_week4\_exercise1, use a combination of **string and path** **controls** to set the location and filename of an output data file.

Use the path control to select the folder to which some data will be saved (we’ll add saving in the next exercise), and the string control to set the filename. Append a formatted date stamp to the filename/path, so that the data file is saved as: /path/to/outputdata/filename\_YYMMDD.

Add a front panel path indicator to show where the data are saved.

Be selective about **where** you build the output filepath / name (i.e. inside vs outside of the while loop), as this will affect the output filename.

Remember to use descriptive names for the objects in the VI.

Save-as “labview101\_week4\_exercise2” into the LabVIEW101\_library.

Exercise 3: Add a save to measurement file function to the “labview101\_week4\_exercise2” VI . Save two numeric signals.

DON’T link the path/filename created above to the save to measurement file function just yet. Instead, run the VI to **manually** select the save file name / destination.

Afterwards, use the file path created in the previous exercise to automatically set the filename of the output data file.

Finally, modify the VI to include **three saving options**: Write to measurement file, save to CSV file, don’t save. Use a combo box/e-num and case structure for this – pay attention to the order of items in the e-num. Again, be selective about where you place the e-num on the block diagram (inside vs outside the while loop).

**Add a Boolean** control to toggle data saving on/off in the first 2 cases (best to use a local variable).

Consider using local variables to make it easier to share the numeric signals to the saving functions.

Save-as “labview101\_week4\_exercise3” into the LabVIEW101\_library.

Exercise 4: Add a property node to **control the upper and lower y-limits** on the plot in your VI.

Use front panel controls to set these so that the values can be adjusted while the program is running.

**Change the precision** of the front panel controls to allow you to increase/decrease the plot scales in 0.25 increments (right click > properties; change Data entry and Display format). Consider whether the y-scale should be set to auto-scale.

Save-as “labview101\_week4\_exercise4” into the LabVIEW101\_library.

Exercise 5: Spend some time cleaning up the VI, and adjusting the appearance. Use decorations, comments, descriptive (yet concise) names for each variable, etc.

Save “labview101\_week4\_exercise4”.

**Arrays & Clusters (& flat sequences)**

* Open /examples and functions/examples/week5\_array.vi
* Arrays
  + Initializing array
  + Growing array: “build array”, “insert into array”
  + Sub-setting array: get values from array element-wise; obtain subset
  + Recall: saving array (write to delimited file, or save to meas. file)
* Clusters: similar to arrays, but can hold various different elements
  + Must initialize with all elements
  + Bundle and unbundle (if items are named, can (un)bundle by name)
    - Side note: can even name string/numeric constants on BD
* Flat sequences:
  + To perform operations in discrete sections.
  + Can add before / after or remove
  + Note: can replace with stacked sequence to save space (right click on flat sequence > replace with stacked sequence)

**Reading from file**

Open /examples & functions/examples/week5\_read-file.vi

Set directories:

* in first 2 examples, use files created last week
* in 3rd example, use /examples & functions/output examples/example\_file.csv
* Read .lvm file (case 0)
  + Note how Read from measurement file (RFMF) is setup
* Read .csv file (case 1)
  + NOTE how RFMF is setup differently than in the last example.
* Read mixed string & num (case 2)

**Next week (last class before end of year):**

Some common errors

Intro to error handling

Running loops concurrently

Reviewing what we’ve learned so far