**Assignment 5: Due – Sunday, July 21 at 9 PM**

Deliverables:

* In your GitHub git repo, right in the HW04 subdirectory (no other subdirectories), push:
  + problem1.c or problem1.cpp (only one of these, not both)
  + problem2.c or problem2.cpp (only one of these, not both)
  + problem3.cu
  + assignment05.pdf

Grading:

* Building: Running on Euler >> bash build.sh from within the HW09 directory should produce the following executables directly in the HW09 directory:
  + problem1
  + problem2
  + problem3
* Running environments (modules) on Euler:
  + Problem1: gcc/recommended
  + Problem2: pgi/18.10 (recall that OpenACC will require a GPU)
  + Problem3: cuda/10.0

Notes:

* Use [this bash script](https://uwmadison.box.com/s/90zxl6s1y122qkizg7znrs0g7d15n4i6) to build your executables
* Use Slurm and follow cluster policy.
* Do not copy solutions off the internet. Please be honest.
* Do **not** submit your executable files

-----------------------------------------------------------------------------------------------------

**Problem 1. (25 Points)**

1. Use OpenMP to evaluate the integral

.

For reference, the value provided by MATLAB for this integral is *I* = 32.121040666358. To approximate the value of  use the following extended Simpson's rule:

In the approximation above, , , , and . This value of  goes to say that you divide the interval  in 106 subintervals when evaluating .

We will run your solution like:

>> ./problem1 npartitions

Remarks:

* npartitions is an integer that indicates the number of sub-intervals used to evaluate the integral of interest. Since you are going to know npartitions, the latter can be used to figure out the value of *h* in Simpson’s formula above.
* The output of the executable problem1 should contain two pieces of information, which are written to a file called problem1.out using the following format:
  + On the first line, a double, the absolute value of the difference between the MATLAB value and the value you obtain using Simpson’s formula
  + On the second line of the file, a double, the amount of time in milliseconds required to compute the integral
* Perform a scaling analysis to understand how your results change as npartitions gets increasingly larger. Use for this analysis the values . Report the results in a log plot in the PDF file assignment05.pdf and indicate in the caption that the results are for OpenMP/Problem 1.

**Problem 2. (25 Points)**

Redo Problem 1 all over again but use OpenACC instead of OpenMP.

**Problem 3. (25 Points)**

Redo Problem 1 all over again but use thrust instead of OpenMP.

**Problem 4 (25 Points)**

In one paragraph in file assignment05.pdf, comment on two things:

* Rank the level of difficulty in producing the code required to compute the integral of interest with OpenMP, OpenACC and thrust. Which approach was the most/least convenient?
* Comment on the speed of execution for each of the three approaches used to approximate *I*.