

Practical Exercise #11

Directive-Based GPU Programming

Exercises

1. Find compilers suitable for your local GPU system. Are both OpenACC and OpenMP compilers available in your case? If not, do you have access to any systems that would allow you to try out these pragma-oriented languages?
2. Run the stream triad examples from the OpenACC/StreamTriad and/or OpenMP/StreamTriad directories on a local GPU-based development system. You can find these directories at <https://github.com/EssentialsofParallelComputing/Chapter11>.
3. Compare your results from Exercise 2 with the BabelStream results at <https://uob-hpc.github.io/BabelStream/results/>. For the stream triad, the bytes moved are $3 * nsize * sizeof(datatype)$.
4. Modify the OpenMP data region mapping in Listing 11.16 to reflect actual array usage in the compute kernels.
5. Implement the mass sum example from Listing 11.4 in OpenMP.
6. For arrays x and y of size 20,000,000, find the maximum radius in the arrays using both OpenMP and OpenACC. Initialize the arrays to double-precision values that increase linearly from 1.0 to $2.0e7$ for the x array and decrease linearly from $2.0e7$ to 1.0 for the y array.