

## Assignment #6

Due: 5/24/2011

## Assignment #6

**Due: 5/24/2011**

This program will continue your saga to find collision pairs of spheres. You are to write programs to:

- 1) Get a user-specified N, D, Seed from command line arguments.
- 2) On host, specify random seed (Seed) using srand(), and then generate N random spheres (3 coordinate components + radius), centers of these spheres should be in the region of [-D:D, -D:D, -D:D] and each sphere is numbered incrementally starting from 0. (of course, you need to use dynamic arrays to store all the data!)
- 3) Write a global function that takes arguments 1) N, the number of spheres, and 2) data for spheres. (i.e. x, y, z, r). This function will then launch a kernel to count the number of pairs of spheres that collides.
  - **There are several challenges on writing this function (to be discussed in the next lecture!) Better start working on this one early!**

## Discuss

1. Discuss the efficiency of storing N sphere data using various data layout strategies. Which of the following gives the best performance?
  1. Option 1: double xyz[N][3], r[N];
  2. Option 2: double xyzr[N][4];
  3. Option 3: double x[N], y[N], z[N], r[N];
  4. You may also explore other possibilities (e.g. using struct & class)
2. Is CUDA version faster or slower? How much faster is your CUDA version than the pure CPU version?

## Note

- All discussions need to consider different program sizes (e.g. N=10, 50, 100, 200, 500, 1000, 2000, 5000, 10000, ...)
- All discussions should be facilitated by charts, and then tell stories from these charts.
- All your work must reside in folder HW06 in your home directory in our cluster system (ssh into 140.118.5.6:222)
- All your programs must be made by simply typing make in HW06 folder (i.e. you must write a Makefile)
- For evaluating performance or efficiency, use the timing class that is provided to you:
  - /home/courses/stopWatch.h, /home/courses/stopWatch.o