Name: _____

Directions: MAKE SURE TO COPY YOUR ANSWERS TO A SEPARATE SHEET FOR SENDING ME AN ELECTRONIC COPY LATER.

1. (10) Fill in the blank with a term from our course: Compared to OpenMP, code written in CUDA tends to have smaller/finer _____.

2. (20) Consider the CUDA example of transforming an adjacency matrix in Sec. 5.12, which we will compare to the similar OMP example in Sec. 4.13. In the latter, consider line 54. State which line (specify the line number) in the CUDA version this corresponds to, or if there is none, why none is needed.

3. (20) Consider the CUBLAS example, Sec. 5.17.1.1. Suppose we simply want to compute the product of the top row of our matrix with the specified vector. Show how to change line 33 to accomplish this.

4. Consider the Mutual Outlinks example, Sec. 5.8. There each thread does a lot of work, but here we'll change it so that each thread will handle exactly one row of the adjacency matrix.

Line 51 will replace the 192 by **tperb** ("threads per block"), taken from the command line.

(a) (20) What specific restriction must we impose on the user in terms of the grid and block sizes?

(b) (15) State which two lines must be changed in the kernel, and show what they should be changed to.

(c) (15) When I first compiled the (unchanged) program, I got an error message, "atomicAdd undefined" (but no other errors). What likely error did I make?

Solutions:

1. granularity

2. No corresponding line. The second kernel call depends on results from the first, and CUDA will notice that will require a wait.

3.

```
cublasSgemv('n',1,n,1.0,dm,n,dm,n,0.0,drs,1);
```

4.a Must have **n = nblk * tperb**, so don't have more or fewer threads than matrix rows.

4.b Change line 14 to

```
i = me;
```

Delete line 19.

4.c Forgot to include **-arch=s_11** in the compile line. This is needed because **atomicAdd()** is a function usable only on models 1.1 and above.