

1. How many floating operations are being performed in your vector add kernel?

There are n number of floating point operations where n is the size of the vector. $A + B = C$ is one floating point operation.

2. How many global memory reads are being performed by your kernel?

There are $2n$ number of global memory reads where n is the size of the vector. This is due to the thread needing to pull the value for A and B which takes two memory reads.

3. How many global memory writes are being performed by your kernel?

There are n number of global memory writes where n is the size of the vector. This is due to the thread only needing to write to memory once when saving C .

4. Describe what possible optimizations can be implemented to your kernel to achieve a performance speedup

Saving the vectors to shared or constant memory will increase performance as the time taken to read to and write to memory decreases

5. Name three applications of vector addition.

Adding a filter in 3-D rendering. Physics engine. CAD