

CS 475/575 -- Spring Quarter 2022
Project #7A

OpenCL / OpenGL Particle System

120 Points

Due: June 7 -- 23:59:59 -- No Bonus Days

Han-Hsing Pao

933651943

pao@oregonstate.edu

1. A web link to the video showing your program in action -- be sure your video is Unlisted.
https://media.oregonstate.edu/media/t/1_ff3aocvl

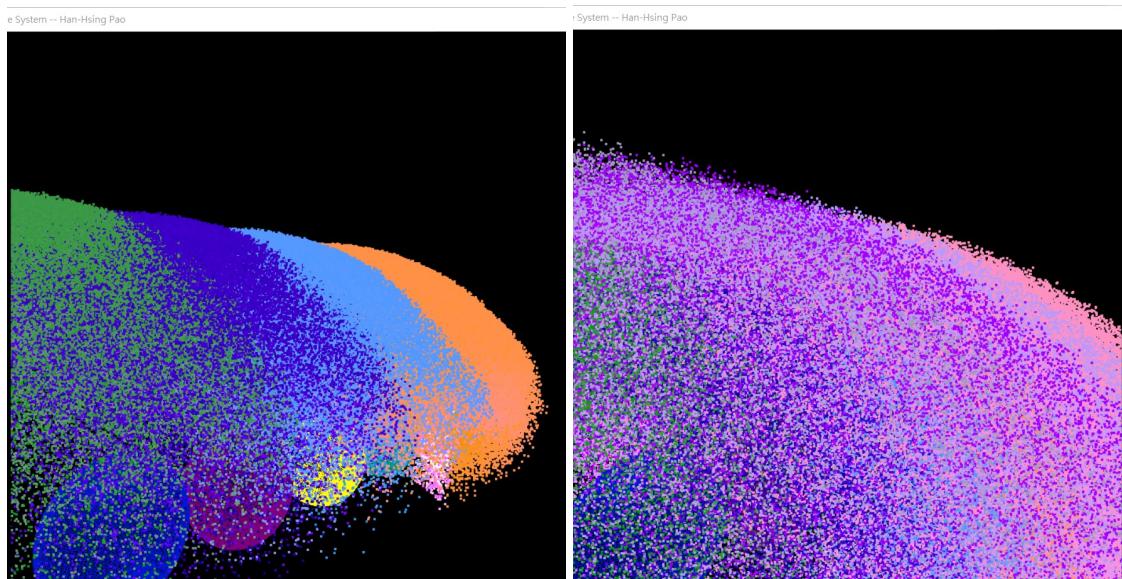
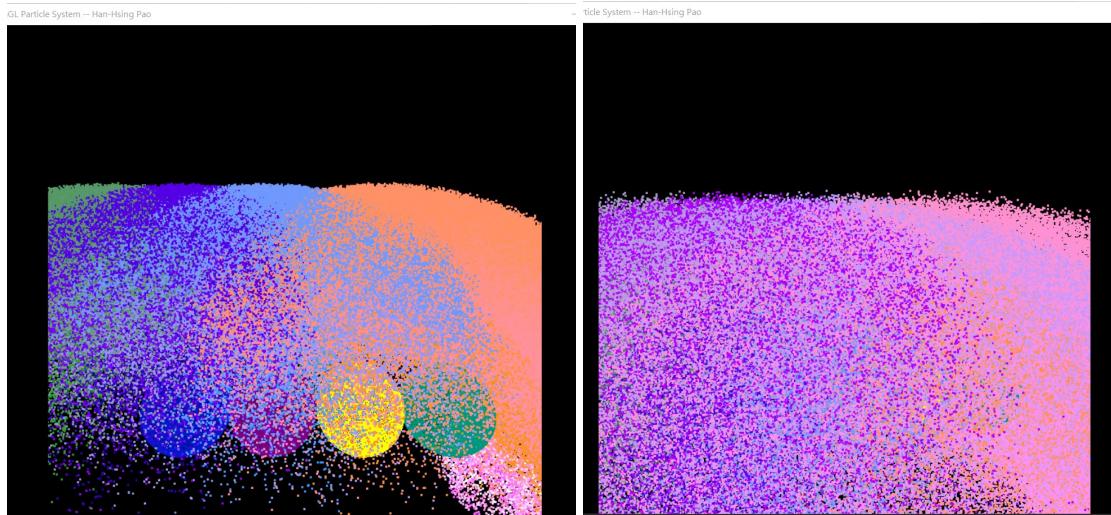
2. What machine you ran this on

I ran on my PC

3. What predictable dynamic thing did you do with the particle colors (random changes are not good enough)

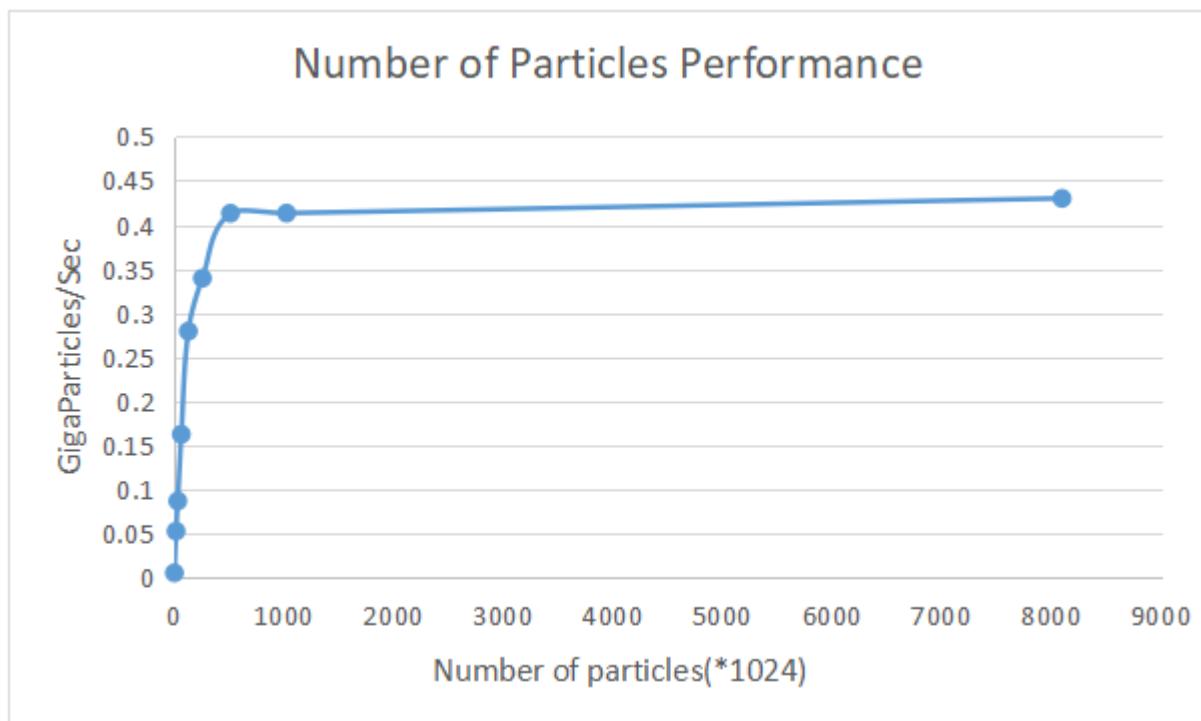
There are four sphere buffers, when a particle hits a sphere, the particle changes color and it gradually turns pink.

4. Include at least one screen capture image of your project in action



5. Show the table and graph

	A	B
1	Number of particles(*1024)	GigaParticles/Sec
2	1	0.00604486
3	16	0.05326398
4	32	0.08742796
5	64	0.16290331
6	128	0.27965009
7	256	0.33965276
8	512	0.41318307
9	1024	0.41334595
10	8096	0.43



6. What patterns are you seeing in the performance curve?

Given a fixed local workgroup size, performance increases and decreases with particle size. When the number of particles increases, the performance also increases.

7. Why do you think the patterns look this way?

When the number of particles is minimal, the performance is low, but as the number of particles grows, so does the performance. This demonstrates that the application is efficient since the GPU requires enough work to overcome all of the overhead associated with its setup.

8. What does that mean for the proper use of GPU parallel computing?

If the particle count is low, it's not worth executing on the GPU, and not enough work is done on the GPU to overcome the overhead of all setups, according to proper GPU parallelism

usage. Only when the total number of particles reaches a certain threshold can GPU parallel computing overcome the setup's overhead.