Use commands to run the CPU version:

```
nvcc −o cgl cgl.cu
./cgl
```

Use commands to run the GPU version:

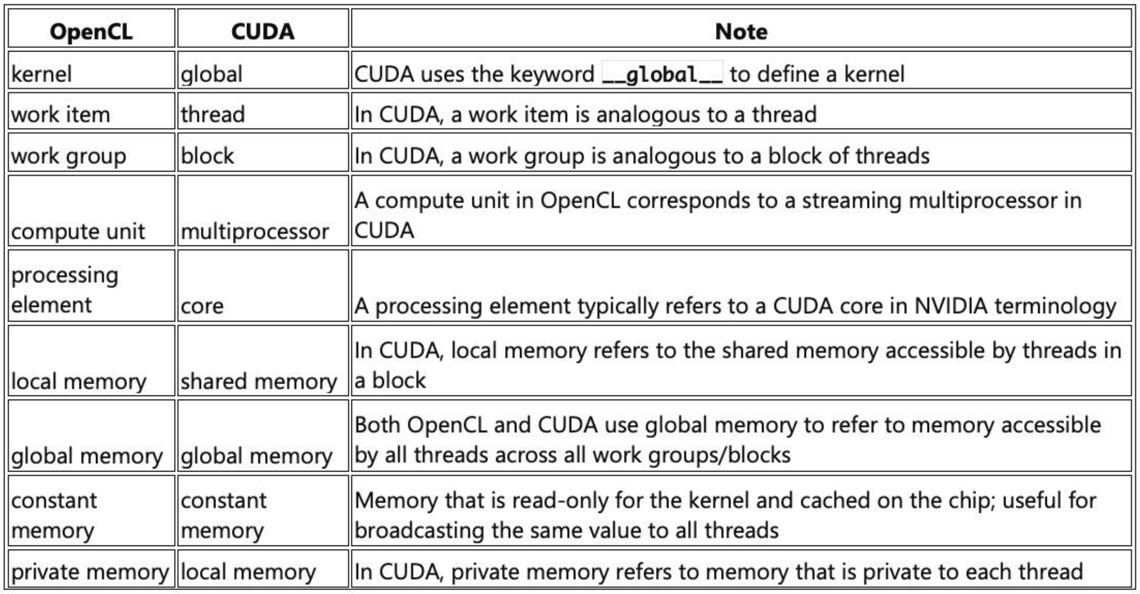
```
nvcc −o cglgpu cglgpu.cu
./cglgpu
```

Use commands to run the GPUSharedMemory version:

```
nvcc −o cglsharedmemory cglsharedmemory.cu
./cglsharedmemory
```

Execution time record in milliseconds:

Generations	CPU	GPU	GPUSharedMemory
1	25.901659	0.207470	0.215080
2	25.422342	0.121471	0.125195
3	25.456868	0.120967	0.124178
4	25.472965	0.121100	0.123643
5	25.440151	0.120298	0.124425
6	25.454412	0.120301	0.124305
7	25.399603	0.120704	0.123775
8	25.528057	0.120571	0.124066
9	25.505825	0.119893	0.124647
10	25.596776	0.119969	0.124297



OpenCL	CUDA	Note
kernel	global	CUDA uses the keyword `__global__` to define a kernel
work item	thread	In CUDA, a work item is analogous to a thread
work group	block	In CUDA, a work group is analogous to a block of threads
compute unit	multiprocessor	A compute unit in OpenCL corresponds to a streaming multiprocessor in CUDA
processing element	core	A processing element typically refers to a CUDA core in NVIDIA terminology
local memory	shared memory	In CUDA, local memory refers to the shared memory accessible by threads in a block
global memory	global memory	Both OpenCL and CUDA use global memory to refer to memory accessible by all threads across all work groups/blocks
constant memory	constant memory	Memory that is read-only for the kernel and cached on the chip; useful for broadcasting the same value to all threads
private memory	local memory	In CUDA, private memory refers to memory that is private to each thread

OpenCL
CUDA
Note

kernel

global

CUDA uses the keyword __global__ to define a kernel

work item

thread

In CUDA, a work item is analogous to a thread

work group

block

In CUDA, a work group is analogous to a block of threads

compute unit

multiprocessor

A compute unit in OpenCL corresponds to a streaming multiprocessor in CUDA

processing element

core

A processing element typically refers to a CUDA core in NVIDIA terminology

local memory

shared memory

In CUDA, local memory refers to the shared memory accessible by threads in a block

global memory

global memory

Both OpenCL and CUDA use global memory to refer to memory accessible by all threads across all work groups/blocks

constant memory

constant memory

Memory that is read-only for the kernel and cached on the chip; useful for broadcasting the same value to all threads

private memory

local memory

In CUDA, private memory refers to memory that is private to each thread