**고급소프트웨어실습1**

**12주차 과제**

**20191621 이민영**

1. 실습

주어진 블록 크기에 대해서 nf를 변화시켜가면서 시간을 측정한 후, 왜 그러한 결과/차이가 산출되었는지 분석하세요.

<Block = 64>

|  |
| --- |
|  |
| nf = 1 |
|  |
| nf = 4 |
|  |
| nf = 16 |
|  |
| nf = 64 |
|  |
| nf = 256 |
|  |
| nf = 1024 |

<Block = 128>

|  |
| --- |
|  |
| nf = 1 |
|  |
| nf = 4 |
|  |
| nf = 16 |
|  |
| nf = 64 |
|  |
| nf = 256 |
|  |
| nf = 1024 |

<Block = 1024>

|  |
| --- |
|  |
| nf = 1 |
|  |
| nf = 4 |
|  |
| nf = 16 |
|  |
| nf = 64 |
|  |
| nf = 256 |
|  |
| nf = 1024 |

block size를 64, 128, 1024로 변화시켜가면서, nf의 크기를 점차 늘려보았을 때, block size에 상관없이 nf의 크기를 증가시켰을 경우에 포함되는 원소가 많기 때문에 수행시간이 증가하게 되고, 각각 CPU, GPU NO SHARED, GPU SHARED를 비교해보았을 때는 CPU가 가장 오래 걸린다. GPU NO SHARED와 GPU SHARED를 비교해보았을 때, nf가 1, 4와 같이 매우 작을 경우에는 수행 시간이 비슷하거나 GPU SHARED가 더 오래 걸리는 경우도 존재하였다. 그러나 nf가 커져서 수행해야하는 원소들이 많이 포함되었을 때는 shared memory를 공유하는 thread이 많아져서 공통으로 사용하는 부분이 많아지게 되어 NO SHARED보다 훨씬 빠르게 수행되는 것을 확인할 수 있었다. block size를 증가 시킴에 따라서, 특히 SHARED GPU의 경우에 block size가 작을 때 보다 훨씬 더 빨리 수행되는 것을 확인할 수 있었다.

1. 과제

다양한 thread block 크기에 따른 수행 시간을 비교하세요.

|  |  |  |
| --- | --- | --- |
|  |  |  |
| CPU : 28.00 ms | no-shared GPU : 5.02 ms | shared GPU : 3.81 ms |
| block size =32 | | |
|  |  |  |
| CPU : 29.00 ms | no-shared GPU : 5.40 ms | shared GPU : 3.26 ms |
| block size = 88 | | |
|  |  |  |
| CPU : 28.00 ms | no-shared GPU : 5.09 ms | shared GPU : 2.77 ms |
| block size = 1024 | | |

CPU, no-shared GPU, shared GPU를 사용했을 때 시간은 CPU가 가장 오래걸리고, shared GPU가 가장 빠른 시간에 수행된 것을 확인할 수 있었다. 또한 block size를 늘렸을 때, shared GPU의 경우 시간이 점점 빨라진 것을 확인할 수 있었으며, 이는 shared GPU의 경우에는 global memory에 비해서 접근 속도가 매우 빠르고 한 블록 내에 있는 thread들이 서로 공유할 수 있는 shared memory를 사용했기 때문이다.