

Assignment 3. Improving Access to Global Memory

CUDA Device Specification

```
C:\masterhpc\hpc heterogeneous prog>deviceinfo
->CUDA Platform & Capabilities
Name: GeForce GTX 750 Ti
totalGlobalMem: 4096.00 MB
sharedMemPerBlock: 48.00 KB
regsPerBlock (32 bits): 65536
warpSize: 32
memPitch: 2097152.00 KB
maxThreadsPerBlock: 1024
maxThreadsDim: 1024 x 1024 x 64
maxGridSize: 2147483647 x 65535
totalConstMem: 64.00 KB
major.minor: 5.0
clockRate: 1110.35 MHz
textureAlignment: 512
deviceOverlap: 1
multiProcessorCount: 5
C:\masterhpc\hpc heterogeneous prog>
```

Problem 1 - Monolithic Kernel

<https://github.com/mahsanchez/masterhpc/blob/master/sqArrSkel.cu>

	Block				
N	32	64	128	256	512
50,000000	0.011s	0.006s	0.007s	0.006s	0.006s
100,000000	0.020s	0.012s	0.013s	0.012s	0.012s
200,000000	0.037s	0.024s	0.024s	0.023s	0.024s

Problem 2 – Block Cyclic Version

https://github.com/mahsanchez/masterhpc/blob/master/sqArrSkel_bc.cu

	Block				
N	32	64	128	256	512
50,000000	0.007000s	0.007000s	0.007000s	0.007000s	0.007000s
100,000000	0.015000s	0.014000s	0.015000s	0.015000s	0.014000s
200,000000	0.030000s	0.028000s	0.028000s	0.028000s	0.028000s

Screenshot of the case of execution for 50,000,000 and 32 tasks per block and k 500

```
VS2015 x64 x86 Cross Tools Command Prompt
C:\masterhpc\hpc heterogeneous prog>nvprof sqArrSkel_bc 50000000 32 500
Time taken by Host: 0.445000s
==8844== NUPROF is profiling process 8844, command: sqArrSkel_bc 50000000 32 500
Time taken by GPU: 0.007000s
Successful Sum
==8844== Profiling application: sqArrSkel_bc 50000000 32 500
==8844== Warning: Found 24 invalid records in the result.
==8844== Warning: This can happen if device ran out of memory or if a device kernel was stopped due to an assertion.
==8844== Profiling result:
   Type  Time(%)   Time      Calls      Avg      Min      Max   Name
GPU activities:  50.48%  91.222ms      1  91.222ms  91.222ms  91.222ms  [CUDA memcpy HtoD]
                45.50%  82.212ms      1  82.212ms  82.212ms  82.212ms  [CUDA memcpy DtoH]
                4.02%   7.2682ms      1   7.2682ms  7.2682ms  7.2682ms  square(float*, int, int)
API calls:      59.25%  372.59ms      1  372.59ms  372.59ms  372.59ms  cudaMalloc
                27.66%  173.94ms      2   86.968ms  82.530ms  91.406ms  cudaMemcpy
                10.50%   66.048ms      1   66.048ms  66.048ms  66.048ms  cuDevicePrimaryCtxRelease
                1.22%   7.6825ms      1   7.6825ms  7.6825ms  7.6825ms  cudaDeviceSynchronize
                0.66%   4.1567ms      1   4.1567ms  4.1567ms  4.1567ms  cudaFree
                0.50%   3.1704ms      1   3.1704ms  3.1704ms  3.1704ms  cuDeviceGetName
                0.11%   710.99us      72   9.9850us    540ns   337.08us  cuDeviceGetAttribute
                0.06%   380.29us      1   380.29us  380.29us  380.29us  cuModuleUnload
                0.02%   123.16us      1   123.16us  123.16us  123.16us  cudaLaunchKernel
                0.00%   15.126us      1   15.126us  15.126us  15.126us  cuDeviceGetPCIBusId
                0.00%   11.344us      1   11.344us  11.344us  11.344us  cuDeviceTotalMem
                0.00%   5.4020us      3   1.8000us    540ns   3.7820us  cuDeviceGetCount
                0.00%   2.1610us      2   1.0800us    540ns   1.6210us  cuDeviceGet
```

Problem 3 – Block Distribution

https://github.com/mahsanchez/masterhpc/blob/master/sqArrSkel_bd.cu

	Block				
N	32	64	128	256	512
50,000000	0.030000s	0.034000s	0.034000s	0.034000s	0.034000s
100,000000	0.059000s	0.065000s	0.064000s	0.064000s	0.065000s
200,000000	0.102000s	0.127000s	0.122000s	0.121000s	0.127000s

Screenshot of the case of execution for 50,000,000 and 32 tasks per block and k 500

```

C:\masterhpc\hpc heterogeneous prog>sqArrSkel_bd 50000000 32 500
Time taken by Host: 0.443000s
Time taken by GPU: 0.029000s
Successful Sum

C:\masterhpc\hpc heterogeneous prog>nvprof sqArrSkel_bd 50000000 32 500
Time taken by Host: 0.443000s
==8300== NUPROF is profiling process 8300, command: sqArrSkel_bd 50000000 32 500
Time taken by GPU: 0.030000s
Successful Sum
==8300== Profiling application: sqArrSkel_bd 50000000 32 500
==8300== Warning: Found 36 invalid records in the result.
==8300== Warning: This can happen if device ran out of memory or if a device kernel was stopped due to an assertion.
==8300== Profiling result:
Type      Time(%)      Time      Calls      Avg      Min      Max      Name
GPU activities: 45.17%  91.616ms    1  91.616ms  91.616ms  91.616ms  [CUDA memcpy HtoD]
              40.54%  82.226ms    1  82.226ms  82.226ms  82.226ms  [CUDA memcpy DtoH]
              14.29%  28.974ms    1  28.974ms  28.974ms  28.974ms  square_block(float*, int, int)
API calls:    55.85%  367.76ms    1  367.76ms  367.76ms  367.76ms  cudaMalloc
              26.45%  174.15ms    2  87.074ms  82.528ms  91.620ms  cudaMemcpy
              12.01%  79.004ms    1  79.004ms  79.004ms  79.004ms  cuDevicePrimaryCtxRelease
              4.46%  29.396ms    1  29.396ms  29.396ms  29.396ms  cudaDeviceSynchronize
              0.56%  3.711ms     1  3.711ms  3.711ms  3.711ms  cudaFree
              0.47%  3.115ms     1  3.115ms  3.115ms  3.115ms  cuDeviceGetName
              0.11%  749.77us    61  12.291us   540ns  354.90us  cuDeviceGetAttribute
              0.06%  384.61us    1  384.61us  384.61us  384.61us  cuModuleUnload
              0.02%  111.82us    1  111.82us  111.82us  111.82us  cudaLaunchKernel
              0.00%  15.125us    1  15.125us  15.125us  15.125us  cuDeviceGetPCIBusId
              0.00%  11.344us    1  11.344us  11.344us  11.344us  cuDeviceTotalMem
              0.00%  5.4020us    3  1.8000us   540ns  4.3220us  cuDeviceGetCount
              0.00%  1.6200us    1  1.6200us  1.6200us  1.6200us  cuDeviceGet

C:\masterhpc\hpc heterogeneous prog>_

```

Problem 4 - Read-only Data Cache

https://github.com/mahsanchez/masterhpc/blob/master/sqArrSkel_cm.cu

	Block				
N	32	64	128	256	512
50,000000	0.011000s	0.007000s	0.006000s	0.006000s	0.006000s
100,000000	0.020000s	0.012000s	0.013000s	0.012000s	0.012000s
200,000000	0.037000s	0.024000s	0.024000s	0.024000s	0.024000s

Screenshot of the case of execution for 50,000,000 and 32 tasks per block and k 500

```

C:\masterhpc\hpc heterogeneous prog>sqArrSkel_cm 50000000 32 500
Time taken by Host: 0.443000s
Time taken by GPU: 0.011000s
Successful Sum

C:\masterhpc\hpc heterogeneous prog>nvprof sqArrSkel_cm 50000000 32 500
Time taken by Host: 0.442000s
==11324== NUPROF is profiling process 11324, command: sqArrSkel_cm 50000000 32 500
Time taken by GPU: 0.011000s
Successful Sum
==11324== Profiling application: sqArrSkel_cm 50000000 32 500
==11324== Warning: Found 19 invalid records in the result.
==11324== Warning: This can happen if device ran out of memory or if a device kernel was stopped due to an assertion.
==11324== Profiling result:
Type      Time(%)      Time      Calls      Avg      Min      Max      Name
GPU activities: 56.96%  135.19ms    1  135.19ms  135.19ms  135.19ms  [CUDA memcpy DtoH]
              38.61%  91.643ms    1  91.643ms  91.643ms  91.643ms  [CUDA memcpy HtoD]
              4.43%  10.512ms    1  10.512ms  10.512ms  10.512ms  square(float*, float const *, int)
API calls:    54.41%  394.94ms    2  197.47ms  16.786ms  378.15ms  cudaMalloc
              31.35%  227.55ms    2  113.77ms  91.819ms  135.73ms  cudaMemcpy
              11.00%  80.416ms    1  80.416ms  80.416ms  80.416ms  cuDevicePrimaryCtxRelease
              1.51%  10.938ms    1  10.938ms  10.938ms  10.938ms  cudaDeviceSynchronize
              1.00%  7.2904ms    2  3.6452ms  3.5760ms  3.7143ms  cudaFree
              0.45%  3.2833ms    1  3.2833ms  3.2833ms  3.2833ms  cuDeviceGetName
              0.10%  737.36us    77  9.5760us   540ns  348.96us  cuDeviceGetAttribute
              0.08%  563.96us    1  563.96us  563.96us  563.96us  cuModuleUnload
              0.02%  115.60us    1  115.60us  115.60us  115.60us  cudaLaunchKernel
              0.00%  16.745us    1  16.745us  16.745us  16.745us  cuDeviceGetPCIBusId
              0.00%  11.084us    1  11.084us  11.084us  11.084us  cuDeviceTotalMem
              0.00%  5.9420us    3  1.9800us   540ns  4.3220us  cuDeviceGetCount
              0.00%  2.1600us    2  1.0800us   540ns  1.6200us  cuDeviceGet

C:\masterhpc\hpc heterogeneous prog>

```

Problem 5

Solution Block Cyclic registered the best performance due potentially to a better usage of global access memory. Increasing the number of threads on memory bound problem do not provides any improvement in performance or hardware usability. Monolithic Kernels implemented in Problem 1 and Problem 4 registered the second and third best response time but it shows that using wisely the cache lines is one of the best way to address performance whenever access to global memory.

Block Distribution memory access pattern registered the worst performance/response time.