### Assignment 3. Improving Access to Global Memory

# **CUDA Device Specification**

```
C:\masterhpc\hpcheterogeneousprog>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
maxThreadsPim: 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
```

#### 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

```
C:\masterhpc\hpcheterogeneousprog\nuprof 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
Successfull Sum
=8844= Profiling application: sqArrSkel_bc 50000000 32 500

Successfull Sum
=8844= Profiling round 24 invalid records in the result.
=8844= Varning: Tound 24 invalid records in the result.
=8844= Varning: This can happen if device ran out of memory or if a device kernel was stopped due to an assertion.

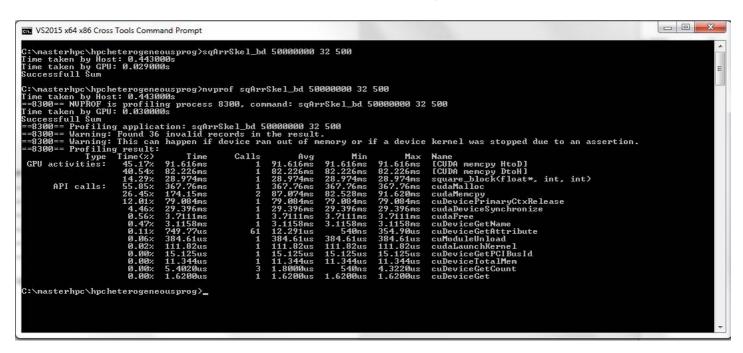
Type Time(x) Time Calls Avg Min Max
GPU activities: 50.48v 91.222ms 1 91.222ms 91.222ms 12.22ms [CIDA memcpy btol] 45.50v 82.212ms 1 82.212ms 82.21
```

#### 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



#### 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:\masterhpr\huchstrogeneousprog\sqfrrSkel_cn 50000000 32 500

Time taken by Host: 9.442000s
Successfull Sun

C:\masterhpr\huchstrogeneousprog\nuprof sqfrrSkel_cn 50000000 32 500

Time taken by GPU: 8.011000s
Successfull Sun

C:\masterhpr\huchstrogeneousprog\nuprof sqfrrSkel_cn 50000000 32 500

Time taken by Host: 9.442000s

=11324== NUPROF is profiling process 11324, command: sqfrrSkel_cn 50000000 32 500

Time taken by GPU: 0.011000s
Successfull Sun

C:\masterhpr\huchstrogeneousprog\nuprof sqfrrSkel_cn 50000000 32 500

Time taken by GPU: 0.011000s

Successfull Sun

Successfull Sun

Fine taken by GPU: 0.011000s

Successfull Sun

Successfull Sun
```

# 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.