

Questions

1)

Assume that a kernel is launched with 1000 thread blocks each of which has 512 threads. If a variable is declared as a shared memory variable, how many versions of the variable will be created through the lifetime of the execution of the kernel?

There will be 1,000 versions of the variable, or one per block. Since shared memory is shared between all threads of the same block, it doesn't matter how many threads are in each block, the number of versions of the variable will be the same as the number of blocks.

2)

For our tiled matrix-matrix multiplication kernel, if we use a 32X32 tile, what is the reduction of memory bandwidth usage for input matrices A and B?

The reduction in bandwidth usage will be 1/32 of the original usage. Each row and column stored in shared memory will be reused 32 times, reducing the global DRAM traffic by the same factor.

3)

For the tiled single-precision matrix multiplication kernel as shown in lecture, assume that the tile size is 32X32 and the system has a DRAM burst size of 128 bytes. How many DRAM bursts will be delivered to the processor as a result of loading one A-matrix tile by a thread block?

There will be 32 DRAM bursts delivered to the processor. Each single-precision value is 4 bytes, so each burst delivers 32 values, or one row of the tile. So, it will take 32 bursts to deliver the whole tile.

12b.ii)

How many warps are running? Based on the Tesla T4 hardware, can you explain this number in terms of hardware resources?

I am not using the Tesla T4 hardware, but rather a GTX 3070 on my own machine. There are 2048 warps allocated. The grid dimension is 16x16 and the block dimension is 16x16. With 32 threads per warp, there are 8 warps per block. With 16x16 blocks, there are $16 \times 16 \times 8 = 2048$ total warps.

Timing Results

	Test Image 1		Test Image 2		Test Image 3		Test Image 4	
	Timer 1 (ms)	Timer 2 (ms)	Timer 1 (ms)	Timer 2 (ms)	Timer 1 (ms)	Timer 2 (ms)	Timer 1 (ms)	Timer 2 (ms)
Global Mem	9.086	0.245	35.785	0.951	86.110	1.625	1497.504	25.022
Static Mem	8.927	0.367	35.111	1.007	86.581	2.510	1499.965	40.648
Dynamic Mem	9.700	0.358	35.428	0.705	86.245	2.976	1501.708	26.906

NSight Compute Profile

Parameter	Kernel	Test Image 1	Test Image 2	Test Image 3	Test Image 4
Kernel Execution Time (seconds)	Global Memory	1.82E-06	2.14E-06	1.89E-06	1.79E-06
	Shared Memory	3.07E-06	2.75E-06	3.39E-06	2.72E-06
Kernel Instruction Intensity (Inst/Byte)	Global Memory	1.38E-03	7.18E-03	1.25E-02	1.23E-02
	Shared Memory	7.52E-03	1.52E-02	1.61E-02	4.12E-03

Appendix A: Result Images

Image 1

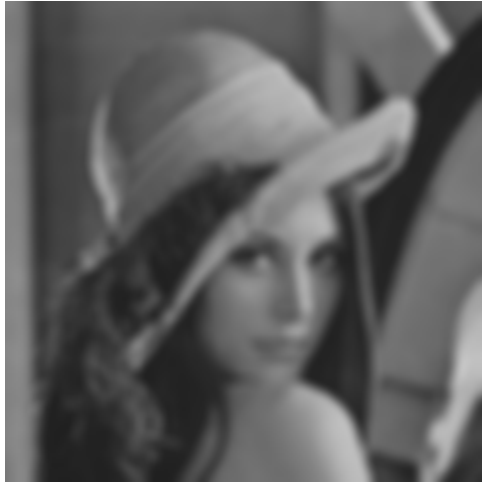


Image 2



Image 3



Image 4



Appendix B: CUDA Kernel Debug

Process: [18052] hw2.exe | Lifecycle Events | Thread: [1] CUDA Thread | Stack Frame: blurKernelDynamicMemory

hw2_starter.cu | (Global Scope) | blurKernelDynamicMemory

```
96 extern __shared__ float s_blurFilter[];
97
98 __global__ void blurKernelDynamicMemory
99 {
100     int col = blockIdx.x * blockDim.x +
101     int row = blockIdx.y * blockDim.y +
102     int filterPadding = (BLUR_FILTER_WIDTH
103
104     // Copy filter coefficients from gl
105     if (threadIdx.x < 9 && threadIdx.y < 9)
106         s_blurFilter[threadIdx.y * BLUR_F
107
108     __syncthreads();
109
110     // Apply the filter to the image
111     if (col < imgDim && row < imgDim) {
112         float pixFloatVal = 0.0;
113         float pixNormalizeFactor = 0.0;
114         int pixVal = 0;
115         int pixels = 0;
116
117         // Get the weighted average of
118         int curRow = 0;
119         int curCol = 0;
120         for (int blurRow = -filterPadding
121             for (int blurCol = -filterPad
122                 curRow = row + blurRow;
123                 curCol = col + blurCol;
124
125                 // Verify we have a valid image pixel
126                 if (curRow > -1 && curRow < imgDim && curCol > -1 && curCol < imgDim) {
127                     pixFloatVal += (float)(imgData[curRow * imgDim + curCol]) * s_blurFilter[blurRow * BLUR_FILTER_WIDTH + blurCol];
128                     pixNormalizeFactor += s_blurFilter[blurRow * BLUR_FILTER_WIDTH + blurCol]; // Accumulate a factor to normalize by
```

Lanes

Lane	Thread Index	Status	PC	Exception
0	(0, 4, 0)	Breakpoint	00000007 00bbaeb0	None
1	(1, 4, 0)	Active	00000007 00bbaeb0	None
2	(2, 4, 0)	Active	00000007 00bbaeb0	None
3	(3, 4, 0)	Active	00000007 00bbaeb0	None
4	(4, 4, 0)	Active	00000007 00bbaeb0	None
5	(5, 4, 0)	Active	00000007 00bbaeb0	None
6	(6, 4, 0)	Active	00000007 00bbaeb0	None
7	(7, 4, 0)	Active	00000007 00bbaeb0	None
8	(8, 4, 0)	Active	00000007 00bbaeb0	None
9	(9, 4, 0)	Active	00000007 00bbaeb0	None
10	(10, 4, 0)	Active	00000007 00bbaeb0	None
11	(11, 4, 0)	Active	00000007 00bbaeb0	None
12	(12, 4, 0)	Active	00000007 00bbaeb0	None
13	(13, 4, 0)	Active	00000007 00bbaeb0	None
14	(14, 4, 0)	Active	00000007 00bbaeb0	None

Warp Info

Context	SM Version	Grid ID	Shader Info
1ecf28700d0	00000006	00000001	CTA: (0, 0, 0), Thread: (0, 0, 0)
1ecf28700d0	00000006	00000001	CTA: (0, 0, 0), Thread: (0, 2, 0)
1ecf28700d0	00000006	00000001	CTA: (0, 0, 0), Thread: (0, 4, 0)
1ecf28700d0	00000006	00000001	CTA: (0, 0, 0), Thread: (0, 6, 0)
1ecf28700d0	00000006	00000001	CTA: (0, 0, 0), Thread: (0, 8, 0)
1ecf28700d0	00000006	00000001	CTA: (0, 0, 0), Thread: (0, 10, 0)
1ecf28700d0	00000006	00000001	CTA: (0, 0, 0), Thread: (0, 12, 0)
1ecf28700d0	00000006	00000001	CTA: (0, 0, 0), Thread: (0, 14, 0)
1ecf28700d0	00000006	00000001	CTA: (0, 1, 0), Thread: (0, 0, 0)
1ecf28700d0	00000006	00000001	CTA: (0, 1, 0), Thread: (0, 2, 0)
1ecf28700d0	00000006	00000001	CTA: (0, 1, 0), Thread: (0, 4, 0)
1ecf28700d0	00000006	00000001	CTA: (0, 1, 0), Thread: (0, 6, 0)
1ecf28700d0	00000006	00000001	CTA: (0, 1, 0), Thread: (0, 8, 0)
1ecf28700d0	00000006	00000001	CTA: (0, 1, 0), Thread: (0, 10, 0)
1ecf28700d0	00000006	00000001	CTA: (0, 1, 0), Thread: (0, 12, 0)

Process: [17912] hw2.exe | Lifecycle Events | Thread: [1] CUDA Thread | Stack Frame: blurKernelDynamicMemory

hw2_starter.cu | (Global Scope) | blurKernelDynamicMemory(unsigned char * imgData, unsigned char * imgOut, float * blurFilter, int imgDim)

```
87 }
88 // Write our new pixel value out
89 imgOut[row * imgDim + col] = (u
90
91 // Blur kernel #2 - device shared memem
92 extern __shared__ float s_blurFilter[];
93
94 __global__ void blurKernelDynamicMemory
95 {
96     int col = blockIdx.x * blockDim.x +
97     int row = blockIdx.y * blockDim.y +
98     int filterPadding = (BLUR_FILTER_WIDTH
99
100     // Copy filter coefficients from gl
101     if (threadIdx.x < 9 && threadIdx.y < 9)
102         s_blurFilter[threadIdx.y * BLUR_F
103
104     __syncthreads();
105
106     // Apply the filter to the image
107     if (col < imgDim && row < imgDim) {
108         float pixFloatVal = 0.0;
109         float pixNormalizeFactor = 0.0;
110         int pixVal = 0;
111         int pixels = 0;
112
113         // Get the weighted average of the surrounding pixels using the gaussian blur filter
114         int curRow = 0;
115         int curCol = 0;
116         for (int blurRow = -filterPadding; blurRow < filterPadding + 1; ++blurRow) {
117             for (int blurCol = -filterPadding; blurCol < filterPadding + 1; ++blurCol) {
118                 curRow = row + blurRow;
119                 curCol = col + blurCol;
```

Lanes

Lane	Thread Index	Status	PC	Exception
0	(0, 0, 0)	Breakpoint	00000007 00bbaeb0	None
1	(0, 0, 0)	Invalid	00000007 00bbaeb0	None
2	(0, 0, 0)	Invalid	00000007 00bbaeb0	None
3	(0, 0, 0)	Invalid	00000007 00bbaeb0	None
4	(0, 0, 0)	Invalid	00000007 00bbaeb0	None
5	(0, 0, 0)	Invalid	00000007 00bbaeb0	None
6	(0, 0, 0)	Invalid	00000007 00bbaeb0	None
7	(0, 0, 0)	Invalid	00000007 00bbaeb0	None
8	(0, 0, 0)	Invalid	00000007 00bbaeb0	None
9	(0, 0, 0)	Invalid	00000007 00bbaeb0	None
10	(0, 0, 0)	Invalid	00000007 00bbaeb0	None
11	(0, 0, 0)	Invalid	00000007 00bbaeb0	None
12	(0, 0, 0)	Invalid	00000007 00bbaeb0	None
13	(0, 0, 0)	Invalid	00000007 00bbaeb0	None
14	(0, 0, 0)	Invalid	00000007 00bbaeb0	None

Warp Info

Context	SM Version	Grid ID	Shader Info	Threads	PC	Active Mask	Valid Mask	Status	Details
117b2c900d0	00000006	00000001	CTA: (0, 0, 0), Thread: (0, 0, 0)	00000000 00000000 00000000 00000000	00000007 00bbaeb0	00000001	00000001	Breakpoint	New

Appendix C: NSight Metrics

Global Memory Kernel

```
[7864] image1_global.exe@127.0.0.1
blurKernelGlobalMemory(unsigned char *, unsigned char *, float *, int, int), 2022-Oct-25 16:54:28, Context 1, Stream 7
Section: Command line profiler metrics
-----
```

dram__bytes.sum	Kbyte	21.38
dram__sectors_read.sum	sector	668
dram__sectors_write.sum	sector	0
smsp__cycles_elapsed.sum	cycle	491,296
smsp__cycles_elapsed.sum.per_second	cycle/nsecond	269.35
smsp__inst_executed.sum	inst	80

```
-----
```

```
[7916] image2_global.exe@127.0.0.1
blurKernelGlobalMemory(unsigned char *, unsigned char *, float *, int, int), 2022-Oct-25 17:06:17, Context 1, Stream 7
Section: Command line profiler metrics
-----
```

dram__bytes.sum	Kbyte	11.14
dram__sectors_read.sum	sector	308
dram__sectors_write.sum	sector	40
smsp__cycles_elapsed.sum	cycle	577,768
smsp__cycles_elapsed.sum.per_second	cycle/nsecond	269.48
smsp__inst_executed.sum	inst	80

```
-----
```

```
[23020] image3_global.exe@127.0.0.1
blurKernelGlobalMemory(unsigned char *, unsigned char *, float *, int, int), 2022-Oct-25 17:08:32, Context 1, Stream 7
Section: Command line profiler metrics
-----
```

dram__bytes.sum	Kbyte	6.40
dram__sectors_read.sum	sector	200
dram__sectors_write.sum	sector	0
smsp__cycles_elapsed.sum	cycle	508,304
smsp__cycles_elapsed.sum.per_second	cycle/nsecond	269.23
smsp__inst_executed.sum	inst	80

```
-----
```

```
[5604] image4_global.exe@127.0.0.1
blurKernelGlobalMemory(unsigned char *, unsigned char *, float *, int, int), 2022-Oct-25 17:11:30, Context 1, Stream 7
Section: Command line profiler metrics
-----
```

dram__bytes.sum	Kbyte	6.53
dram__sectors_read.sum	sector	204
dram__sectors_write.sum	sector	0
smsp__cycles_elapsed.sum	cycle	485,816
smsp__cycles_elapsed.sum.per_second	cycle/nsecond	271.10
smsp__inst_executed.sum	inst	80

```
-----
```

Shared Memory Kernel

```
[7576] image1_shared.exe@127.0.0.1
blurKernelStaticMemory(unsigned char *, unsigned char *, float *, int, int), 2022-Oct-25 17:14:08, Context 1, Stream 7
Section: Command line profiler metrics
-----
```

dram__bytes.sum	Kbyte	20.22
dram__sectors_read.sum	sector	620
dram__sectors_write.sum	sector	12
smsp__cycles_elapsed.sum	cycle	831,232
smsp__cycles_elapsed.sum.per_second	cycle/nsecond	270.58
smsp__inst_executed.sum	inst	152

```
-----
```

```
[10932] image2_shared.exe@127.0.0.1
blurKernelStaticMemory(unsigned char *, unsigned char *, float *, int, int), 2022-Oct-25 17:15:42, Context 1, Stream 7
Section: Command line profiler metrics
-----
dram__bytes.sum                                Kbyte                                9.98
dram__sectors_read.sum                         sector                               284
dram__sectors_write.sum                       sector                               28
smsp__cycles_elapsed.sum                      cycle                               745,816
smsp__cycles_elapsed.sum.per_second           cycle/nsecond                       271.01
smsp__inst_executed.sum                       inst                                152
-----
```

```
[4672] image3_shared.exe@127.0.0.1
blurKernelStaticMemory(unsigned char *, unsigned char *, float *, int, int), 2022-Oct-25 17:16:44, Context 1, Stream 7
Section: Command line profiler metrics
-----
dram__bytes.sum                                Kbyte                                9.47
dram__sectors_read.sum                         sector                               288
dram__sectors_write.sum                       sector                               8
smsp__cycles_elapsed.sum                      cycle                               852,472
smsp__cycles_elapsed.sum.per_second           cycle/nsecond                       271.83
smsp__inst_executed.sum                       inst                                152
-----
```

```
[2416] image4_shared.exe@127.0.0.1
blurKernelStaticMemory(unsigned char *, unsigned char *, float *, int, int), 2022-Oct-25 17:17:52, Context 1, Stream 7
Section: Command line profiler metrics
-----
dram__bytes.sum                                Kbyte                                36.86
dram__sectors_read.sum                         sector                               1,112
dram__sectors_write.sum                       sector                               40
smsp__cycles_elapsed.sum                      cycle                               740,896
smsp__cycles_elapsed.sum.per_second           cycle/nsecond                       272.39
smsp__inst_executed.sum                       inst                                152
-----
```