Tiled Matrix Multiplication

Questions

1. How many floating operations are being performed in your matrix multiply kernel? Explain.

   Outside the for loop we have one initialization and one assignment, and inside the for loop loop we have $(2 * \texttt{TILE\_WIDTH} + 2) * \texttt{numAColumns / TILE\_WIDTH}$ flops, so we end up performing $2 * \texttt{numARows} * \texttt{numBColumns} * \texttt{numAColumns}$ FLOPs.

2. How many global memory reads are being performed by your kernel? Explain.

   The two if-else statements at the beginning of the for loop read the global memory twice, so one thread performs $2 * \texttt{ceil(numAColumns / TILE\_WIDTH)}$ global memory reads. Thus, the kernel performs $2 * \texttt{numARows} * \texttt{numAColumns} * \texttt{numBColumns}$ global memory reads.

3. How many global memory writes are being performed by your kernel? Explain.

   We have one write at the end of the kernel function, so the kernel performs $\texttt{numARows} * \texttt{numBColumns}$ global memory writes.

4. Describe what further optimizations can be implemented to your kernel to achieve a performance speedup.

   We can optimize the block size to be a multiple of the warp size (32) and use a larger `TILE_WIDTH` to reduce the number of global memory reads and writes.

5. Compare the implementation difficulty of this kernel compared to the previous MP. What difficulties did you have with this implementation?

   I had trouble keeping track of which matrix width to use for setting up the edge cases for the matrix multiplication. In addition, the number of threads per block had to match the tile width!

6. Suppose you have matrices with dimensions bigger than the max thread dimensions. Sketch an algorithm that would perform matrix multiplication algorithm that would perform the multiplication in this case.

   Using grid-stride loops (link),

```
1:  matrixMultGridStride                                  ▷ Global kernel function
2:    rowStart = blockIdx.y × TILE_WIDTH + threadIdx.y    ▷ with same shared memory setup
3:    colStart = blockIdx.x × TILE_WIDTH + threadIdx.x
4:
5:    rowStride = gridDim.y × TILE_WIDTH                  ▷ set up strides
6:    colStride = gridDim.x × TILE_WIDTH
7:
8:    for (row = rowStart; numCRows; ++rowStride) do      ▷ Grid-stride loop
9:      for (col = colStart; numCColumns; ++colStride) do
10:       Cval = 0
11:       ...                                             ▷ Same tiling algorithm as before
```

Algorithm 1: Grid-stride loop for tiled matrix multiplication

OUTPUT

```
[junseo@r40a-09.sif TiledMatrixMultiplication]$ ./tiled_matrix_mult.sh
[TIME][Generic][Importing data and creating memory on host][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 72-82] Elapsed time: 10.0063 ms
The dimensions of A are 16 x 16
The dimensions of B are 16 x 16
[TIME][GPU][Allocating GPU memory.][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 87-93] Elapsed time: 0.257754 ms
[TIME][GPU][Copying input memory to the GPU.][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 95-104] Elapsed time: 0.04098 ms
[TIME][Compute][Performing CUDA computation][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 111-119] Elapsed time: 0.344714 ms
[TIME][Copy][Copying output memory to the CPU][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 121-125] Elapsed time: 0.018813 ms
[TIME][GPU][Freeing GPU Memory][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 127-133] Elapsed time: 0.211362 ms
The solution is correct
[TIME][Generic][Importing data and creating memory on host][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 72-82] Elapsed time: 11.7092 ms
The dimensions of A are 64 x 64
The dimensions of B are 64 x 64
[TIME][GPU][Allocating GPU memory.][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 87-93] Elapsed time: 0.286958 ms
[TIME][GPU][Copying input memory to the GPU.][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 95-104] Elapsed time: 0.077308 ms
[TIME][Compute][Performing CUDA computation][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 111-119] Elapsed time: 0.438718 ms
[TIME][Copy][Copying output memory to the CPU][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 121-125] Elapsed time: 0.040828 ms
[TIME][GPU][Freeing GPU Memory][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 127-133] Elapsed time: 0.153445 ms
The solution is correct
[TIME][Generic][Importing data and creating memory on host][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 72-82] Elapsed time: 13.7811 ms
The dimensions of A are 64 x 128
The dimensions of B are 128 x 64
[TIME][GPU][Allocating GPU memory.][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 87-93] Elapsed time: 0.241301 ms
[TIME][GPU][Copying input memory to the GPU.][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 95-104] Elapsed time: 0.081899 ms
[TIME][Compute][Performing CUDA computation][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 111-119] Elapsed time: 0.380791 ms
[TIME][Copy][Copying output memory to the CPU][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 121-125] Elapsed time: 0.056268 ms
[TIME][GPU][Freeing GPU Memory][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 127-133] Elapsed time: 0.153353 ms
The solution is correct
[TIME][Generic][Importing data and creating memory on host][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 72-82] Elapsed time: 11.2472 ms
The dimensions of A are 112 x 48
The dimensions of B are 48 x 16
[TIME][GPU][Allocating GPU memory.][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 87-93] Elapsed time: 0.2498 ms
[TIME][GPU][Copying input memory to the GPU.][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 95-104] Elapsed time: 0.067243 ms
[TIME][Compute][Performing CUDA computation][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 111-119] Elapsed time: 0.327779 ms
[TIME][Copy][Copying output memory to the CPU][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 121-125] Elapsed time: 0.041162 ms
[TIME][GPU][Freeing GPU Memory][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 127-133] Elapsed time: 0.180934 ms
The solution is correct
[TIME][Generic][Importing data and creating memory on host][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 72-82] Elapsed time: 12.5744 ms
The dimensions of A are 84 x 84
The dimensions of B are 84 x 84
[TIME][GPU][Allocating GPU memory.][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 87-93] Elapsed time: 0.230314 ms
[TIME][GPU][Copying input memory to the GPU.][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 95-104] Elapsed time: 0.08076 ms
[TIME][Compute][Performing CUDA computation][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 111-119] Elapsed time: 0.344323 ms
[TIME][Copy][Copying output memory to the CPU][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 121-125] Elapsed time: 0.044059 ms
[TIME][GPU][Freeing GPU Memory][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 127-133] Elapsed time: 0.184058 ms
The solution is correct
[TIME][Generic][Importing data and creating memory on host][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 72-82] Elapsed time: 15.6087 ms
The dimensions of A are 80 x 99
The dimensions of B are 99 x 128
[TIME][GPU][Allocating GPU memory.][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 87-93] Elapsed time: 0.311499 ms
[TIME][GPU][Copying input memory to the GPU.][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 95-104] Elapsed time: 0.092648 ms
[TIME][Compute][Performing CUDA computation][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 111-119] Elapsed time: 0.349116 ms
[TIME][Copy][Copying output memory to the CPU][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 121-125] Elapsed time: 0.04457 ms
[TIME][GPU][Freeing GPU Memory][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 127-133] Elapsed time: 0.150292 ms
The solution is correct
[TIME][Generic][Importing data and creating memory on host][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 72-82] Elapsed time: 11.3611 ms
The dimensions of A are 67 x 53
The dimensions of B are 53 x 64
[TIME][GPU][Allocating GPU memory.][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 87-93] Elapsed time: 0.255127 ms
[TIME][GPU][Copying input memory to the GPU.][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 95-104] Elapsed time: 0.075223 ms
[TIME][Compute][Performing CUDA computation][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 111-119] Elapsed time: 0.408549 ms
[TIME][Copy][Copying output memory to the CPU][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 121-125] Elapsed time: 0.053768 ms
[TIME][GPU][Freeing GPU Memory][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 127-133] Elapsed time: 0.195668 ms
The solution is correct
[TIME][Generic][Importing data and creating memory on host][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 72-82] Elapsed time: 13.0868 ms
The dimensions of A are 29 x 117
The dimensions of B are 117 x 85
[TIME][GPU][Allocating GPU memory.][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 87-93] Elapsed time: 0.308301 ms
[TIME][GPU][Copying input memory to the GPU.][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 95-104] Elapsed time: 0.078199 ms
[TIME][Compute][Performing CUDA computation][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 111-119] Elapsed time: 0.344332 ms
[TIME][Copy][Copying output memory to the CPU][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 121-125] Elapsed time: 0.039822 ms
[TIME][GPU][Freeing GPU Memory][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 127-133] Elapsed time: 0.158768 ms
The solution is correct
[TIME][Generic][Importing data and creating memory on host][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 72-82] Elapsed time: 12.4133 ms
The dimensions of A are 191 x 19
The dimensions of B are 19 x 241
[TIME][GPU][Allocating GPU memory.][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 87-93] Elapsed time: 0.301441 ms
[TIME][GPU][Copying input memory to the GPU.][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 95-104] Elapsed time: 0.149757 ms
[TIME][Compute][Performing CUDA computation][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 111-119] Elapsed time: 0.408404 ms
[TIME][Copy][Copying output memory to the CPU][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 121-125] Elapsed time: 0.197292 ms
[TIME][GPU][Freeing GPU Memory][/home/warehouse/junseo/cuda-code-repo-joonsuuh/Module4/TiledMatrixMultiplication/solution.cu: 127-133] Elapsed time: 0.157388 ms
The solution is correct
[junseo@r40a-09.sif TiledMatrixMultiplication]$
```

Figure 1: `TiledMatrixMultiplication_Solution` output