گام اول.

```
[Retrix Multiply using CLOM] - Starting...

### Stort part (23.72) | Multiply (23.72) | M
```

# گام دوم.

## راه حل اول: (تابع چک صحت در اسکرینشات مشاهده می شود)

```
roid checkCorrectness(float* v, int matSizeX, int matSizeY) {
                                                        Microsoft Visual Studio Debug Console
                                                                                                        float value = v[0];
                                                       [Matrix Multiply Using CUDA] - Starting...
                                                       GPU Device 0: "GeForce 920MX" with compute capab
      for (int j = 0; j < matSizeY; j++) {
         if (v[i * matSizeY + j] != value)
                                                       ility 5.0
            cout << "Something has gone wrong" << endl;</pre>
                                                       [-] N = 128
            return;
                                                       MatrixA(128,128), MatrixB(128,128)
                                                       Computing result using CUDA Kernel...
                                                       Elapsed time in msec = 8.473696
  cout << "All is right with the world." << endl;
                                                       All is right with the world.
```

### راه حل دوم:

```
// Setup execution parameters
//dim3 threads(n, n, 1);
dim3 threads(n, n, 1);
dim3 threads(n, n, 1);
dim3 threads(n, n, 1);
dim3 grid(n/TILE_WIDTH, 1);

// Create and start timer
printf("Computing result using CUDA Kernel...\n");

// Allocate CUDA events that we'll use for timing
cudaEvent_t start;
error = cudaEventCreate(&start);

If (error != cudaSuccess)

[Matrix Multiply Using CUDA] - Starting...

GPU Device 0: "GeForce 920MX" with compute capab
ility 5.0

[-] N = 12'8

Tile Width: 32

MatrixA(128,128), MatrixB(128,128)

Computing result using CUDA Kernel...
Elapsed time in msec = 2.661344

All is right with the world.
```

#### راه حل سوم:

```
/ Setup execution parameters
                                                                                                                              ×
                                                                      Microsoft Visual Studio Debug Console
//dim3 threads(n, n, 1);
/*dim3 threads(TILE_WIDTH, TILE_WIDTH, 1);
dim3 grid(n/TILE_WIDTH, n/TILE_WIDTH, 1);*/
                                                                     [Matrix Multiply Using CUDA] - Starting...
                                                                     GPU Device 0: "GeForce 920MX" with compute capab
                                                                     ility 5.0
                                                                     [-] N = 128
Tile Width: 4
dim3 threads(16, 16, 1);
dim3 grid(n / (TILE_WIDTH * 16), n / (TILE_WIDTH * 16), 1);
                                                                     MatrixA(128,128), MatrixB(128,128)
// Create and start timer
printf("Computing result using CUDA Kernel...\n");
                                                                     Computing result using CUDA Kernel...
                                                                     CUDA elapsed time in msec = 0.344576
                                                                     All is right with the world.
```

جدول راه حل دو: (زمان و Speedup)

```
Block Size = 4
 -] Grid Size = 64
Tile Width: 32
MatrixA(256,256), MatrixB(256,256)
Computing result using CUDA Kernel...
CUDA elapsed time in msec = 2.202656
All is right with the world.
Serial time difference = 18 ms
All is right with the world.
Speed up: 8.17195
[-] Block Size = 8
[-] Grid Size = 32
Tile Width: 32
MatrixA(256,256), MatrixB(256,256)
Computing result using CUDA Kernel...
CUDA elapsed time in msec = 1.615232
All is right with the world.
Serial time difference = 19 ms
All is right with the world.
Speed up: 11.763
```

روش سوم به صورت پیش فرض به خطا برمیخورد و باید با تغییر دادن تنظیمات tdr از قسمت nsight monitor آن را به کار انداخت

```
[-] Block Size = 32
[-] Grid Size = 128
Tile Width: 32
MatrixA(4096,4096), MatrixB(4096,4096)
Computing result using CUDA Kernel...
CUDA elapsed time in msec = 5940.917480
All is right with the world.
Serial time difference = 404784 ms
All is right with the world.
Speed up: 68.1349
```

جدول راه حل دو: (Occupancy) ها با استفاده از nsight):

.

```
Grid Dimensions
                                                             (64 64 1) 4096
                                                             {4, 4, 1} 16
50.00%
  Block Dimensions
 Occupancy
Registers per Thread
 Static Shared Memory per Block
Dynamic Shared Memory per Block
Shared Memory Configuration Executed
                                                             0 bytes
                                                             0 bytes
FOUR_BYTE_BANK_SIZE
 Local Memory per Thread
                                                             0 bytes
                                                             3.670.016 bytes
 Local Memory
Cache Configuration Requested
                                                             PREFER_NONE
PREFER_SHARED
 Cache Configuration Executed
Cache Configuration Changed
                                                             False
Dynamic Parallelism
```

{32, 32, 1} 1024 {8, 8, 1} 64 100.00% Grid Dimensions Block Dimensions Occupancy
Registers per Thread
Static Shared Memory per Block
Dynamic Shared Memory per Block
Shared Memory Configuration Executed
Local Memory per Thread
Local Memory
Cache Configuration Requested
Cache Configuration Requested Occupancy 0 bytes 0 bytes FOUR\_BYTE\_BANK\_SIZE 0 bytes 3,670,016 bytes PREFER\_NONE Cache Configuration Executed Cache Configuration Changed PREFER\_SHARED

## Configuration

Grid Dimensions {128, 128, 1} 16384 **Block Dimensions** {32, 32, 1} 1024 Occupancy 100.00% Registers per Thread 32

Static Shared Memory per Block 0 bytes Dynamic Shared Memory per Block 0 bytes

Shared Memory Configuration Executed FOUR\_BYTE\_BANK\_SIZE

Local Memory per Thread 0 bytes

Local Memory 3,670,016 bytes Cache Configuration Requested PREFER\_NONE Cache Configuration Executed PREFER SHARED

Cache Configuration Changed False

	4*4, 64*64	8*8, 32*32	32*32, 128*128
Elapsed Time	2.202	1.615	5941
Speedup	8.1719	11.763	68.14
Occupancy	50%	100%	100%

## گام سوم.

زمانهای cuBLAS:

```
-1 N = 32
MatrixA(32,32), MatrixB(32,32)
Elapsed time in msec = 0.02208
-] N = 128
MatrixA(128,128), MatrixB(128,128)
Elapsed time in msec = 0.026464
-] N = 256
MatrixA(256,256), MatrixB(256,256)
Elapsed time in msec = 0.100576
- N = 512
MatrixA(512,512), MatrixB(512,512)
Elapsed time in msec = 0.708352
```

زمانهای روش دوم:

```
[-] N = 32
Tile Width: 32
MatrixA(32,32), MatrixB(32,32)
Computing result using CUDA Kernel...
Elapsed time in msec = 0.014688
All is right with the world.
[-] N = 128
Tile Width: 32
MatrixA(128,128), MatrixB(128,128)
Computing result using CUDA Kernel...
Elapsed time in msec = 0.145344
All is right with the world.
[-] N = 256
Tile Width: 32
MatrixA(256,256), MatrixB(256,256)
Computing result using CUDA Kernel...
Elapsed time in msec = 1.084096
All is right with the world.
[-] N = 512
Tile Width: 32
MatrixA(512,512), MatrixB(512,512)
Computing result using CUDA Kernel...
Elapsed time in msec = 10.172320
All is right with the world.
```

	32	128	256	512
گام دوم	0.014	0.145	1.08	10.17
cuBLAS	0.022	0.026	0.1	0.708
Speedup	1.571	5.577	10.81	14.364