Arquitectura de Computadores Avançada

Grupo 1

Fábio Alves NMEC: 84734

Ricardo Pombeiro NMEC: 71718

GPU USADO

GeForce GTX 1060 6GB

CUDA Driver Version / Runtime Version:	10.1 / 10.1
CUDA Capability Major/Minor version number:	6.1
Total amount of global memory:	5.93 GBytes (6372196352 bytes)
GPU Clock rate:	1785 MHz (1.78 GHz)
Memory Clock rate:	4004 Mhz
Memory Bus Width:	192-bit
L2 Cache Size:	1572864 bytes
Max Texture Dimension Size (x,y,z):	1D=(131072), 2D=(131072,65536), 3D=(16384,16384,16384)
Max Layered Texture Size (dim) x layers:	1D=(32768) x 2048, 2D=(32768,32768) x 2048
Total amount of constant memory:	65536 bytes
Total amount of shared memory per block::	49152 bytes
Total number of registers available per block::	65536
Warp size:	32
Maximum number of threads per multiprocessor:	2048
Maximum number of threads per block:	1024
Maximum sizes of each dimension of a block:	1024 x 1024 x 64
Maximum sizes of each dimension of a grid:	2147483647 x 65535 x 65535
Maximum memory pitch:	2147483647 bytes

Dados coletados usando “checkDeviceInfor” disponibilizado nas aulas práticas.

cryptCuda

Quantidade de SMs (10).	Blocos > 10
Tempos de acesso a memória.	Necessidade de ir buscar dados à memória principal (valores não estão em cache)
Capacidade de transferência.	192 bits = 24 bytes = 6 Nº inteiros

Configuração base

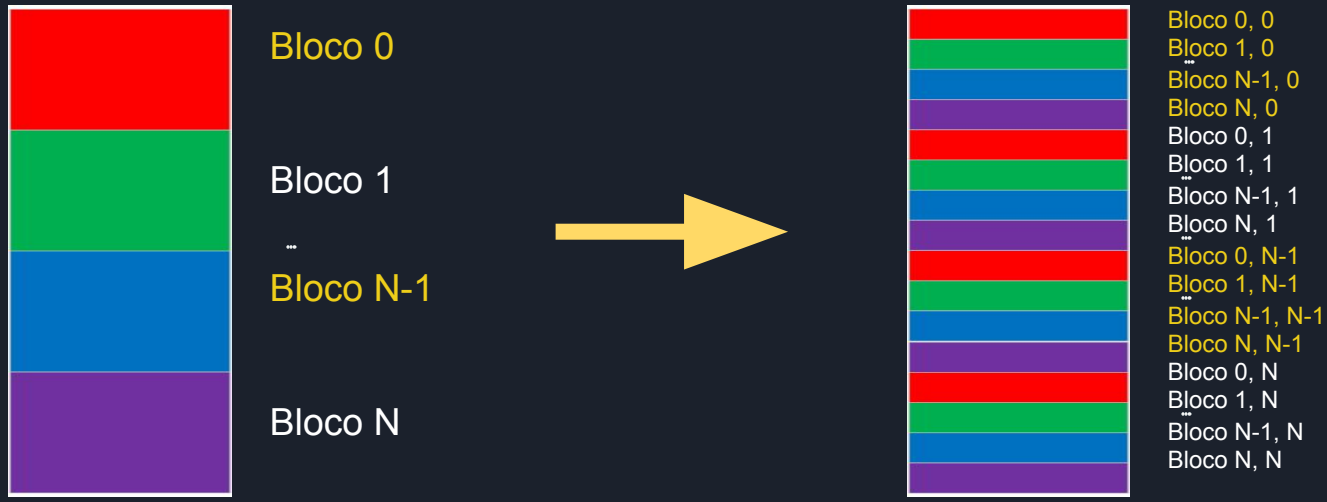
blockDimX	blockDimY	blockDimZ	gridDimX	gridDimY	gridDimZ	GPU Time	CPU Time	GPU Transfer	CPU Transfer
1	1	1	8192	256	1	8.82E-01	3.42E-01	1.83E-01	5.85E-01

cryptCuda

- O melhor desempenho foi com uma *grid size* de (512,128) e um *block size* de (2,16).
- Desempenho do CPU praticamente constante em todos os testes.
- Analisando os resultados, não é possível atingir uma otimização em que a GPU seja mais rápida a transferir dados que o CPU.
- Aumento do tamanho dos blocos resulta numa diminuição da performance.

								Host->Device	Device->Host
lockDimX	blockDimY	blockDimZ	gridDimX	gridDimY	gridDimZ	GPU Time	CPU Time	[illegible]	[illegible]
2	16	1	512	128	1	5.67E-01	3.67E-01	1.81E-01	5.87E-01
2	8	1	512	256	1	5.69E-01	3.73E-01	1.81E-01	5.95E-01
2	16	1	128	512	1	5.75E-01	3.81E-01	1.82E-01	5.93E-01
2	2	1	32768	16	1	5.76E-01	3.68E-01	1.82E-01	5.88E-01
2	8	1	2048	64	1	5.79E-01	3.67E-01	1.82E-01	5.84E-01
2	4	1	32768	8	1	5.80E-01	3.69E-01	1.83E-01	5.99E-01
2	16	1	8192	8	1	5.90E-01	3.66E-01	1.82E-01	5.95E-01
16	2	1	128	512	1	5.93E-01	3.66E-01	1.82E-01	6.00E-01
4	8	1	1024	64	1	6.00E-01	3.67E-01	1.82E-01	5.93E-01
4	4	1	16384	8	1	6.00E-01	3.78E-01	1.82E-01	6.00E-01
16	2	1	64	1024	1	6.03E-01	3.66E-01	1.82E-01	5.87E-01
8	2	1	8192	16	1	6.03E-01	3.66E-01	1.82E-01	5.90E-01
8	4	1	8192	8	1	6.04E-01	3.66E-01	1.81E-01	5.92E-01
8	4	1	1024	64	1	6.06E-01	3.67E-01	1.81E-01	5.92E-01
4	4	1	1024	128	1	6.13E-01	3.66E-01	1.81E-01	5.93E-01
16	4	1	64	512	1	6.28E-01	3.67E-01	1.83E-01	5.95E-01
1	16	1	1024	128	1	6.29E-01	3.67E-01	1.82E-01	5.89E-01
1	16	1	512	256	1	6.29E-01	3.67E-01	1.82E-01	5.89E-01
4	16	1	1024	32	1	6.31E-01	3.88E-01	1.83E-01	6.06E-01
8	16	1	64	256	1	6.33E-01	3.67E-01	1.82E-01	5.95E-01
4	32	1	256	64	1	6.50E-01	3.84E-01	1.82E-01	5.93E-01
4	16	1	256	128	1	6.57E-01	3.67E-01	1.81E-01	5.88E-01
16	16	1	256	32	1	6.73E-01	3.81E-01	1.82E-01	5.95E-01
1	1	1	8192	256	1	8.82E-01	3.42E-01	1.83E-01	5.85E-01

cryptCudaStride



Configuração base

Stride	blockDimX	blockDimY	blockDimZ	gridDimX	gridDimY	gridDimZ	GPU Time	CPU Time	GPU Transfer	CPU Transfer
1	1	1	1	8192	256	1	8.82E-01	3.42E-01	1.83E-01	5.85E-01

cryptCudaStride

Strid	blockDimX	blockDimY	blockDimZ	gridDimX	gridDimY	gridDimZ	GPU Time	CPU Time	Host->Device	Device->Host
2048	64	8	1	4	1024	1	6.60E-02	1.67E+00	1.82E-01	5.87E-01
8192	64	8	1	4	1024	1	6.60E-02	2.36E+00	1.82E-01	5.89E-01
512	128	8	1	2	1024	1	6.66E-02	1.10E+00	1.82E-01	5.87E-01
2048	128	8	1	2	1024	1	6.67E-02	1.72E+00	1.88E-01	6.12E-01
512	64	16	1	64	32	1	6.67E-02	1.10E+00	1.82E-01	5.86E-01
16	16	4	1	64	512	1	6.68E-02	4.77E-01	1.82E-01	5.95E-01
16	16	4	1	128	256	1	6.70E-02	4.72E-01	1.82E-01	5.89E-01
16	16	4	1	2048	16	1	6.70E-02	4.62E-01	1.82E-01	5.86E-01
16	16	4	1	256	128	1	6.71E-02	4.62E-01	1.81E-01	5.90E-01
16	16	4	1	32	1024	1	6.73E-02	4.74E-01	1.82E-01	5.90E-01
16	16	4	1	512	64	1	6.74E-02	4.62E-01	1.82E-01	5.90E-01
512	512	2	1	64	32	1	6.74E-02	1.10E+00	1.82E-01	5.87E-01
1024	1024	1	1	64	32	1	6.75E-02	1.78E+00	1.82E-01	5.94E-01
1024	1024	1	1	16	128	1	6.75E-02	1.68E+00	1.82E-01	5.96E-01
1024	1024	1	1	4	512	1	6.76E-02	1.68E+00	1.83E-01	5.95E-01
1024	1024	1	1	8	256	1	6.76E-02	1.82E+00	1.82E-01	5.91E-01
1024	1024	1	1	2	1024	1	6.77E-02	1.68E+00	1.82E-01	5.97E-01
1024	1024	1	1	32	64	1	6.78E-02	1.67E+00	1.82E-01	6.17E-01
16	16	8	1	256	64	1	6.81E-02	4.67E-01	1.83E-01	5.97E-01
256	256	4	1	64	32	1	6.81E-02	6.58E-01	1.83E-01	6.02E-01
2048	64	16	1	64	32	1	6.82E-02	1.81E+00	1.86E-01	5.99E-01
32	32	32	1	64	32	1	6.89E-02	4.71E-01	1.81E-01	5.85E-01
64	64	16	1	64	32	1	6.92E-02	4.67E-01	1.82E-01	5.93E-01
128	128	8	1	64	32	1	6.94E-02	6.09E-01	1.82E-01	5.86E-01
1	1	1	1	8192	256	1	8.82E-01	3.42E-01	1.83E-01	5.85E-01

Conclusão

1. Para usar o máximo de recursos é necessário possuir um número de blocos superior ao número de SMs (mais de 10 blocos no caso do gpu usado).
2. Com o aumento do stride o tempo de execução do CPU (execução sequencial) aumenta e o do GPU (mais execuções paralelas) diminui. Isto deve-se ao facto do GPU executar vários blocos de forma paralela o que leva a que os dados dos diversos blocos já estejam em cache. Por outro lado o CPU executa de forma sequencial, porém com o stride os dados foram separados , o que leva a ser necessário mais transferências de dados da memória principal para a cache.
3. O melhor stride com um número de blocos <128 é igual ao número do blockx, para valores >128 é 4 vezes superior ao número total de blocos com o exceção de 1024 blocos em que o melhor valor é 512.
4. Os melhores tempos são obtidos com um número elevado de blocos, pela razão mencionada no ponto 2.