GPU컴퓨팅

(공영호 교수님)

Assignment3

학번: 2019202103

학과: 컴퓨터정보공학부

이름: 이은비

**Assignment3**

**<Introduction>**

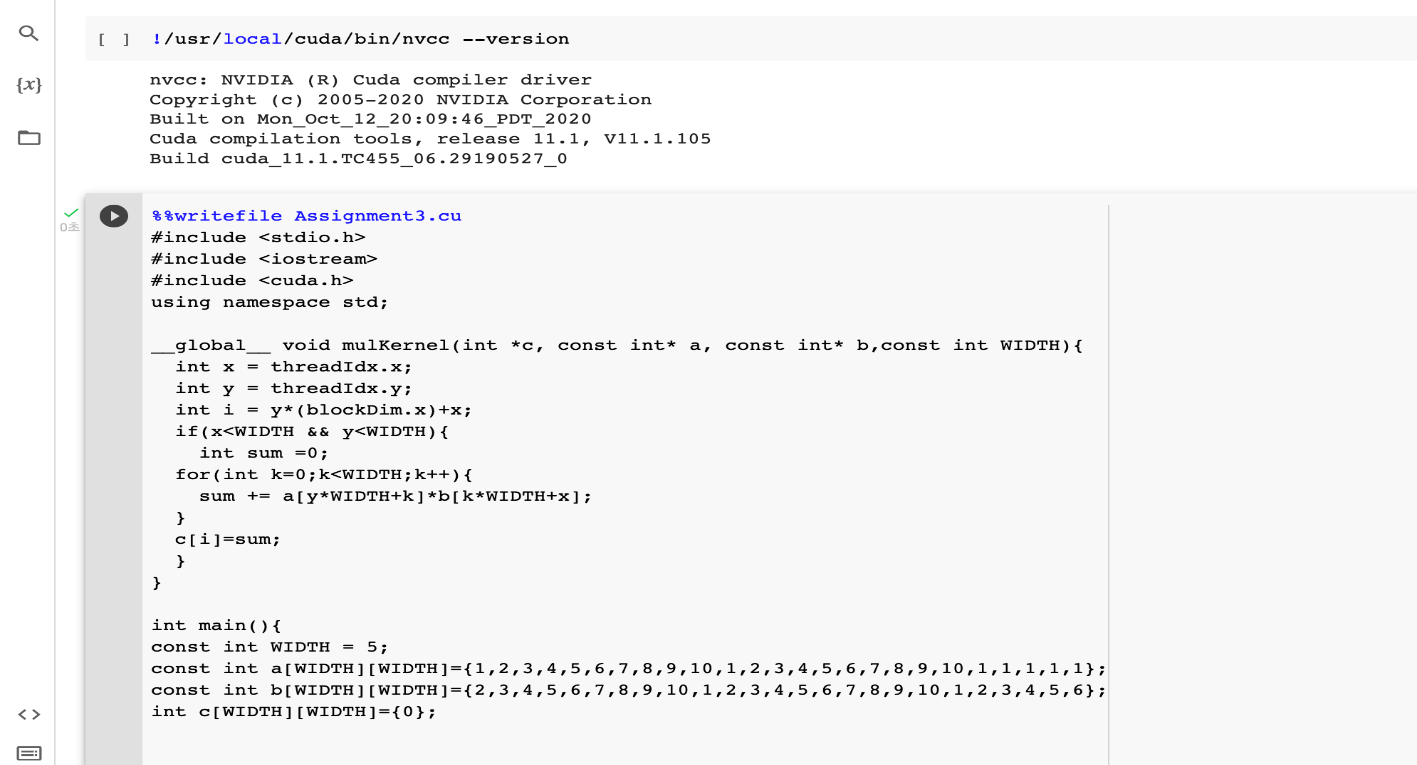
Cuda에서의 hierarchical은 threads(기본단위)->blocks->grids(상위단위)로 구성 되어 있습니다.

이와 같은 특징을 이용하여 2D indexing하여 각각의 Matrix의 원소들,즉 각각의 thread에 접근하여서 병렬적으로 연산을 하는 실습과 과제를 진행합니다. 실습에서는 Matrix addition을 진행하였는데 physical layout을 이해하고 index도 1d처럼 c[i] = y\*WIDTH +x;이며 이를 이용하여 Matrix addition실습을 진행하여 보고, 과제로 Matrix Multiplication을 구현합니다. 이때 matrix multiplication은 host에서 실행할 때 Triple for loop을 사용하는데 triple for loop부터는 CUDA에서 실행 될 때에도 kernel에서 loop연산을 수행하는 점을 적용합니다.

**<Conclusion>**

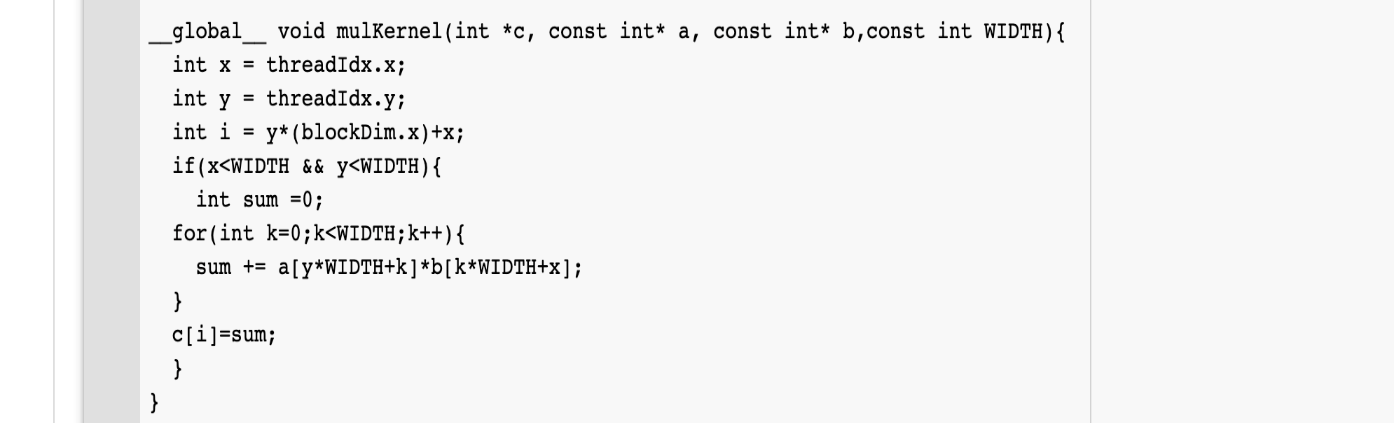
**[Assignment3.cu의 코드내용 및 결과화면]**

**(전체 코드내용)**



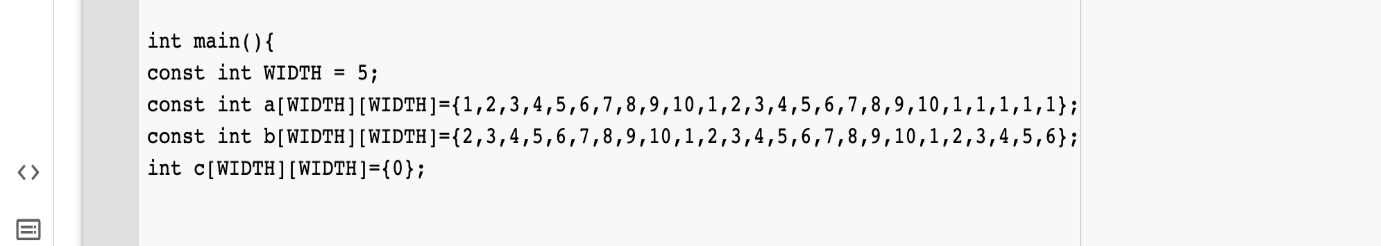


(부분 코드1)



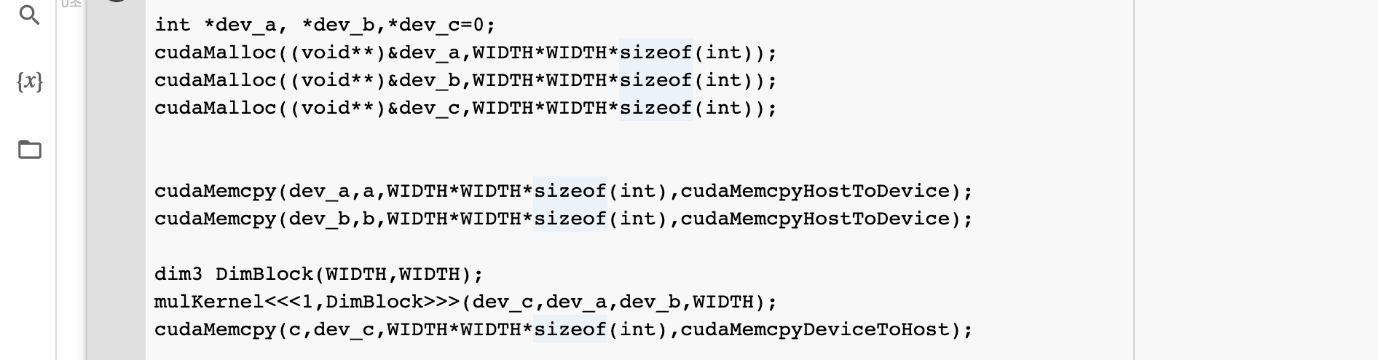
CUDA kernel function을 준비합니다. 이때 \_\_global\_\_은 CPU에서 호출 되어서 GPU에서 실행 될 수 있도록 합니다. 함수의 이름은 mulKernel이며 인수로는 cpu에서 작성된 배열을 copy받아서 곱해지는 대상인 2개의 Matrix인 const int\* a,b와 Matrix Multiplication의 결과 Matrix를 다시 CPU에 copy해줄 int \*c,그리고 loop계산에서 쓰이게 될 Matrix size 인 WIDTH를 const int로 받아옵니다. 또한 함수의 내용을 보면 x와 y는 행과 열인데 현재 Matrix가 정방 행렬이므로 다른 built-in변수를 추가 하지 않고 threadIdx.x,threadIdx.y로만 선언합니다. 또한 배열의 index i는 y\*(blockDim.x)+x;로 1d로 접근할 수 있도록 합니다. 또한 행과 열이 Matrix size를 넘어가지 않는 조건에서 각각의 thread별로 sum 변수로 a의 행과 b의 열을 곱한 결과를 저장하고 각각의 결과값(sum값)을 c Matrix의 각각의 원소(thread)에 할당합니다.

(부분 코드2)



Kernel 힘수 mulkerenl에 전달 하게 될 Matrix a,b를 초기화 하고, 배열의 size인 WIDTH 도 5로 초기화 합니다. Mulkernel의 결과를 전달 받을 c Matrix또한 초기화 합니다..

(부분코드3)



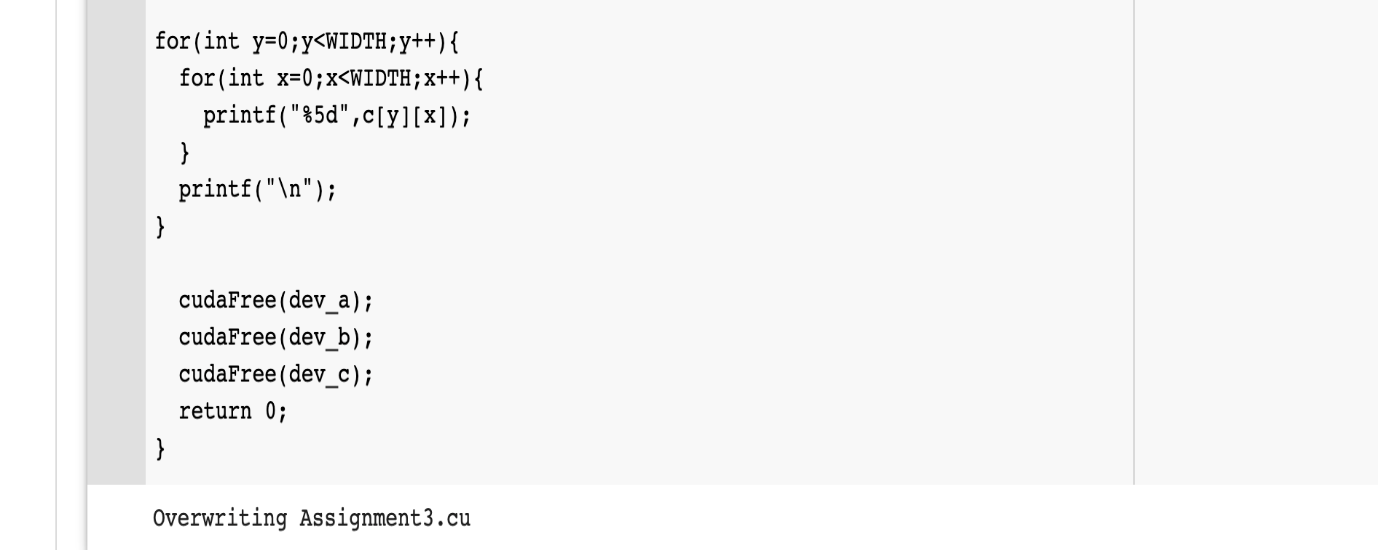
cudaMalloc function을 이용하여 Kernel의 메모리를 할당합니다. 할당되는 사이즈는 전달 받을 사이즈인 matrix의 전체 원소(threads)와 저장될 값인 int형의 size를 각각 곱한 값과 같습니다.

cudaMemcpy function을 이용하여 host의 data를 device로 copy합니다. a,b배열의 값을 각각dev\_a,dav\_b,로 copy합니다

mulKerenl<<<1,WIDTH>>>(dev\_c,dav\_a,dev\_b,WIDTH); 1부터 WIDTH까지 앞서 prepare된 함수(\_\_global\_\_void mulKenel)를 호출합니다.

이후 kernel에서 수행된 결과 값인 cMatrix를 cudaMemcpy로 CPU로 전달합니다.

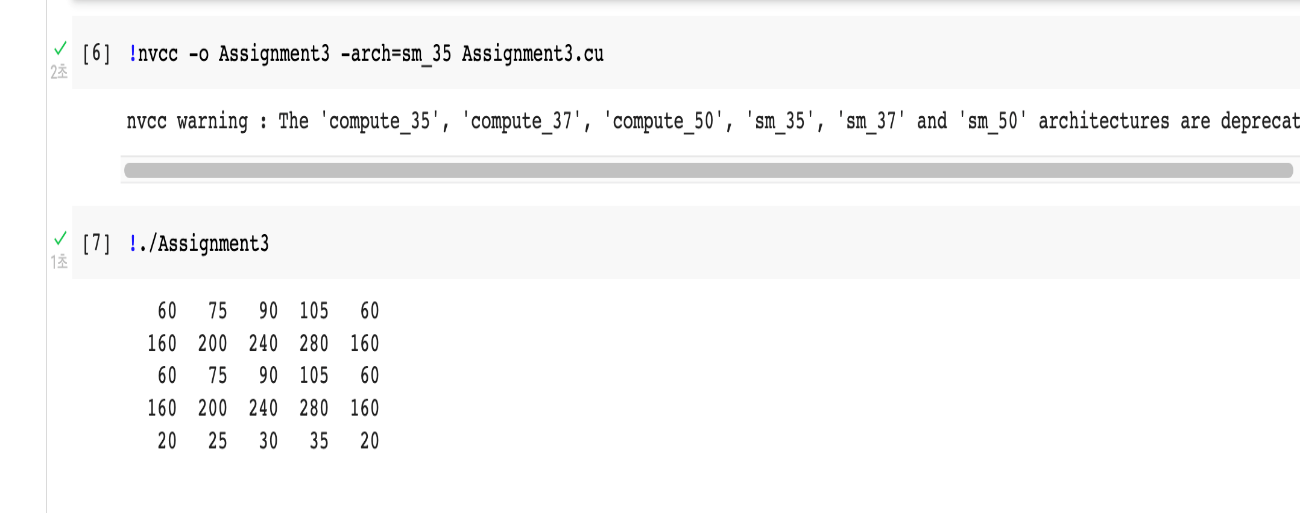
(부분코드4)



CPU에서 Printf함수로 kernel로부터 전달받은 결과인 c Matrix를 2D형식으로 출력합니다.

연산이 종료된 후에는 cudaFree();함수로 앞서 cudaMalloc으로 할당되었던 변수의 메모리를 해제 합니다.

(결과화면)



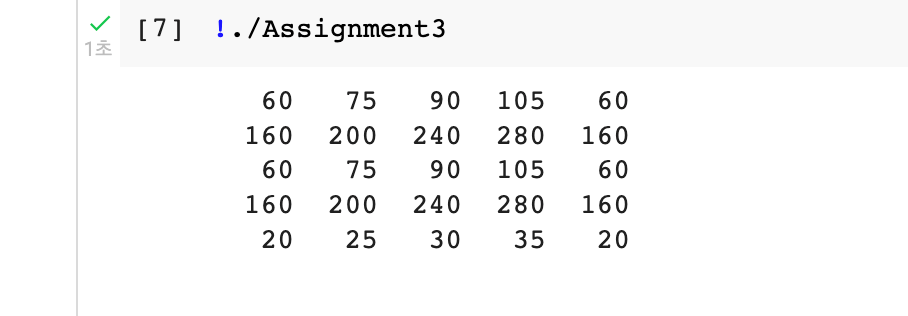
!nvcc -o Assignment3 -arch\_35 Assignment3.cu

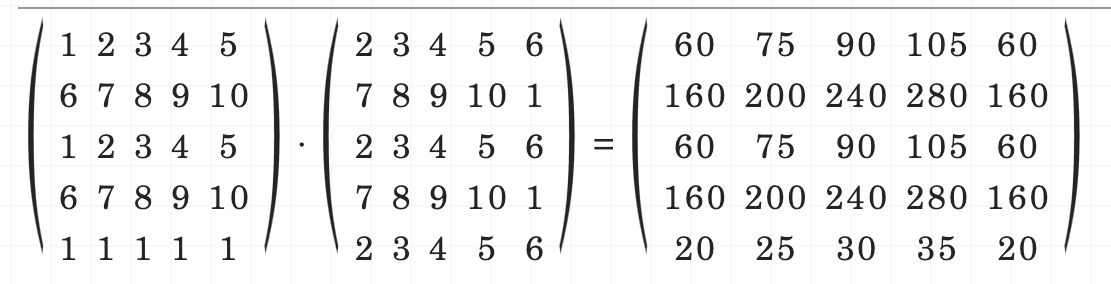
앞서 작성되었던 Assignment3.cu파일을 Assignment3라는 이름의 실행파일로 컴파일합니다.

!./Assignment3

실행파일을 실행시켜서 printf로 작성된 c의 Matrix의 결과가 2d형식으로 출력됩니다.

(결과 matrix값 확인)





앞서 설정한 Matrix의 원소 값을 임의로 설정하여 초기화 한 것이어서 결과를 확인하기 위해 행렬 계산기를 통해 결과값과 비교하여 결과로 나온 Matrix와 같음을 확인하였습니다.

**<Reference>**

강의 자료,행렬계산기 https://matrixcalc.org/ko/