Relazione Homework 2: Matrice

A cura di Sadman Sakib Rahman, Tiziano Natali, Luca Mancini

Matricole: 1632174, 1554933, 1649441

L'IDEA

L'approccio algoritmico utilizzato per effettuare l'estensione da matrici quadrate a matrici rettangolari riadatta l'algoritmo visto a lezione per effettuare la stessa operazione su matrici quadrate. L'idea consiste nell'eseguire in maniera parallela le operazioni di ogni singola cella, e, successivamente, nell'algoritmo che usufruisce della memoria locale, effettuare le stesse operazioni in una sottomatrice interna di dimensioni fisse.

L'IMPLEMENTAZIONE

Per modificare il programma originale della moltiplicazione fra matrici abbiamo innanzitutto aggiunto una nuova variabile globale nel codice host tramite #define per rappresentare la dimensione del lato corto del rettangolo.
Abbiamo quindi cambiato gli indici utilizzati per il calcolo, ricordandoci che una matrice di dimensioni n*k moltiplicata per una matrice di dimensioni k*n, è uguale a una matrice n*n.
Abbiamo inoltre cambiato i valori per l'allocazione in memoria tramite malloc di A e B, e i parametri nel metodo clCreateBuffer dei MEM OBJECT in maniera da rispecchiare la grandezza delle matrici rettangolari.
Nel kernel abbiamo aggiunto una nuova variabile in input, le dimensioni del lato corto. Infine, abbiamo cambiato il codice sequenziale affinché potesse effettuare moltiplicazioni fra matrici rettangolari.

Nel programma con utilizzo della memoria locale dei workgroups, per poter eseguire il programma con matrici rettangolari a prescindere dalla variabile LOCAL_SIZE, abbiamo approssimato le dimensioni dei lati al valore del primo multiplo della LOCAL_SIZE maggiore delle dimensioni originali.

PIATTAFORME

MacBook Pro di Tiziano Natali

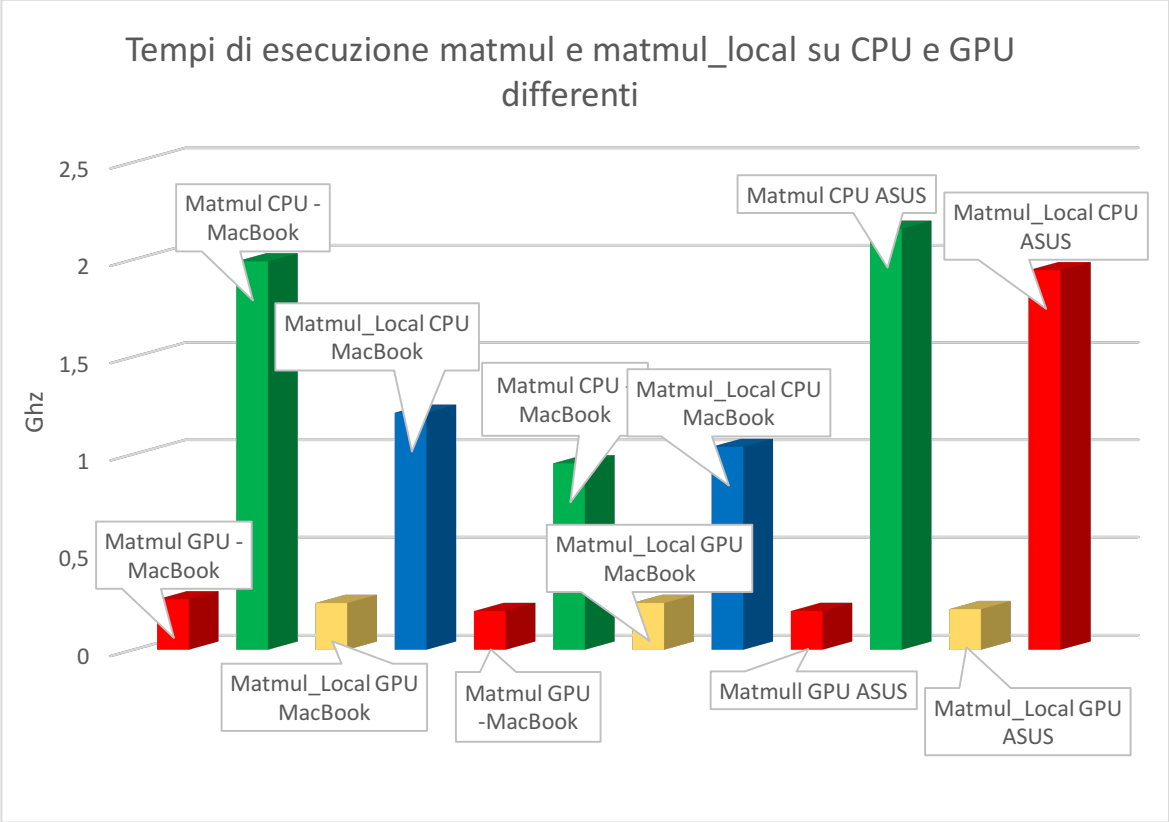
MacBook Pro di S. Sakib Rahman

ASUS TP300LA di Luca Mancini

Intel i5-4210U CPU @ 1.70GHz 2.40
RAM 8.00 GB
Sistema operativ0 64-bit
Model: ASUS TP300LA
WINDOWS 10
GPU- INTEL HD Graphics 4400

DATI SUGLI ESPERIMENTI

Paragone tra tempo di esecuzione parallela e sequenziale su input diversi



CONSIDERAZIONI FINALI

Abbiamo constatato che all'aumentare delle dimensioni dei lati delle matrici il codice che utilizza memoria locale risulta essere il più efficiente. Inoltre all'aumentare del divario fra i lati, l'esecuzione su GPU risulta essere molto più veloce che su CPU.

ISTRUZIONI PER ESECUZIONE DA TERMINALE

cd DIRECTORY/LOCATION_OF_C_FILE

make NAME_OF_C_FILE_WITHOUT_C_EXTENSION

./NAME_OF_C_FILE_WITHOUT_C_EXTENSION

Esempio di esecuzione: