מאיצים חישוביים ומערכות מואצות
046278

דו"ח הגשה – תרגיל בית רטוב 1

מתן צחורי 208936989

עדי צחורי 315374066

חלק 1

a. גרסת ה-Cuda בה השתמשנו היא: $V12.5.4$

```
$ nvcc --version
nvcc: NVIDIA (R) Cuda compiler driver
Copyright (c) 2005-2024 NVIDIA Corporation
Built on Wed_Apr_17_19:19:55_PDT_2024
Cuda compilation tools, release 12.5, V12.5.40
Build cuda_12.5.r12.5/compiler.34177558_0
```

b. שם ה-GPU הוא:
$NVIDIA\ GeForce\ RTX\ 2080$

```
$ nvidia-smi
Fri Jun 14 14:47:41 2024
+-----------------------------------------------------------------------------------------+
| NVIDIA-SMI 550.90.07              Driver Version: 550.90.07      CUDA Version: 12.4     |
|-----------------------------------------+------------------------+----------------------+
| GPU  Name                 Persistence-M | Bus-Id          Disp.A | Volatile Uncorr. ECC |
| Fan  Temp   Perf          Pwr:Usage/Cap |           Memory-Usage | GPU-Util  Compute M. |
|                                         |                        |               MIG M. |
|=========================================+========================+======================|
|   0  NVIDIA GeForce RTX 2080 ...    Off |   00000000:02:00.0 Off |                  N/A |
| 29%   36C    P0             41W /  250W |       1MiB /   8192MiB |      0%      Default |
|                                         |                        |                  N/A |
+-----------------------------------------+------------------------+----------------------+
|   1  NVIDIA GeForce RTX 2080 ...    Off |   00000000:03:00.0 Off |                  N/A |
| 12%   35C    P0             13W /  250W |       1MiB /   8192MiB |      0%      Default |
|                                         |                        |                  N/A |
+-----------------------------------------+------------------------+----------------------+
|   2  NVIDIA GeForce RTX 2080 ...    Off |   00000000:83:00.0 Off |                  N/A |
| 34%   35C    P0             53W /  250W |       1MiB /   8192MiB |      0%      Default |
|                                         |                        |                  N/A |
+-----------------------------------------+------------------------+----------------------+
|   3  NVIDIA GeForce RTX 2080 ...    Off |   00000000:84:00.0 Off |                  N/A |
| 34%   35C    P0             44W /  250W |       1MiB /   8192MiB |      1%      Default |
|                                         |                        |                  N/A |
+-----------------------------------------+------------------------+----------------------+
```

d. מספר ה-SMs הוא: 48
וכן בכל אחד מהם ישנם
64 ליבות $Cuda$.

```
Device 2: "NVIDIA GeForce RTX 2080 SUPER"
  CUDA Driver Version / Runtime Version          12.4 / 12.5
  CUDA Capability Major/Minor version number:    7.5
  Total amount of global memory:                 7967 MBytes (8354398208 bytes)
  (048) Multiprocessors, (064) CUDA Cores/MP:    3072 CUDA Cores
  GPU Max Clock rate:                            1815 MHz (1.81 GHz)
  Memory Clock rate:                             7751 Mhz
  Memory Bus Width:                              256-bit
  L2 Cache Size:                                 4194304 bytes
  Maximum Texture Dimension Size (x,y,z)         1D=(131072), 2D=(131072, 65536), 3D=(16384, 16384, 16384)
  Maximum Layered 1D Texture Size, (num) layers  1D=(32768), 2048 layers
  Maximum Layered 2D Texture Size, (num) layers  2D=(32768, 32768), 2048 layers
  Total amount of constant memory:               65536 bytes
  Total amount of shared memory per block:       49152 bytes
  Total shared memory per multiprocessor:        65536 bytes
  Total number of registers available per block: 65536
  Warp size:                                     32
  Maximum number of threads per multiprocessor:  1024
  Maximum number of threads per block:           1024
  Max dimension size of a thread block (x,y,z): (1024, 1024, 64)
  Max dimension size of a grid size    (x,y,z): (2147483647, 65535, 65535)
  Maximum memory pitch:                          2147483647 bytes
  Texture alignment:                             512 bytes
  Concurrent copy and kernel execution:          Yes with 3 copy engine(s)
  Run time limit on kernels:                     No
  Integrated GPU sharing Host Memory:            No
  Support host page-locked memory mapping:       Yes
  Alignment requirement for Surfaces:            Yes
  Device has ECC support:                        Disabled
  Device supports Unified Addressing (UVA):      Yes
  Device supports Managed Memory:                Yes
  Device supports Compute Preemption:            Yes
  Supports Cooperative Kernel Launch:            Yes
  Supports MultiDevice Co-op Kernel Launch:      Yes
  Device PCI Domain ID / Bus ID / location ID:   0 / 131 / 0
  Compute Mode:
     < Default (multiple host threads can use ::cudaSetDevice() with device simultaneously) >
```

חלק 3

b. במימוש הפונקציות שבהן עושה שימוש הפונקצייה:
$\_\_global\_\_\ void\ process\_image\_kernel(uchar\ *all\_in, uchar\ *all\_out)$
נרצה להשתמש ב-$atomicAdd$ מכיוון שבחלק מהמשלבים בתהליך מספר חוטים עלולים לגשת לאותם מקומות בזיכרון באותו הזמן ועלול להיווצר מצב של $'race\ condition'$ (למשל בפונקציית חישוב ההיסטוגרמה של $Tile$ מסוים).
בחישוב ההיסטוגרמה מתבצעת פעולת $histogram[tid]++$, כאשר בפועל פעולה כזו מורכבת ממספר פעולות קטנות יותר – קריאת תוכן התא, העלאת הערך ב-1, ושמירה חזרה במקום. על מנת לשמור על הקוהרנטיות של התכנית, נרצה לוודא שבמצבים כאלה כל הפעולות מתבצעות כפעולה אחת אטומית, ואין התנגשות בין פעולה של חוט אחד לפעולה של חוט אחר.

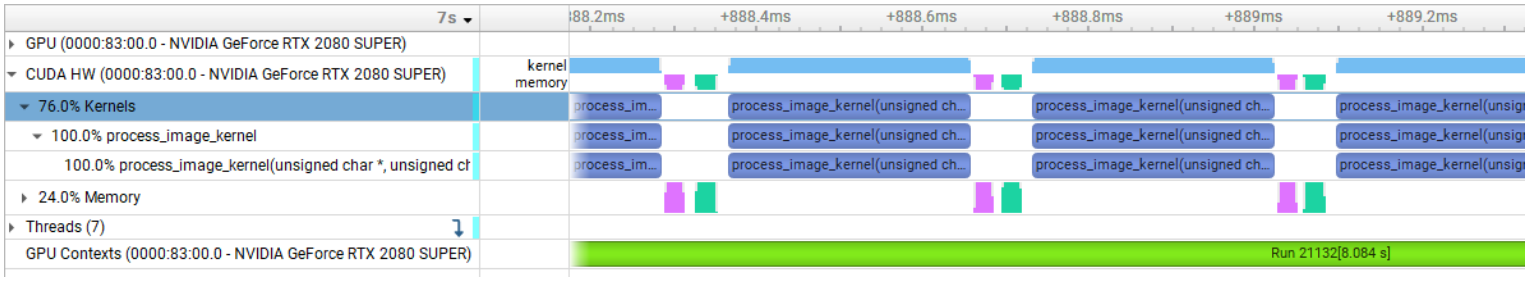
c. הגישות לזכרון הגלובלי מתחברות יחד כיוון שזמן הגישה לזכרון לרוב דורש זמן רב ועלול להוות צוואר בקבוק בחישובים. לשם כך נדרש מנגנון שיאפשר קריאה מהירה ככל הניתן, למשל כמו משיכת מידע רק בו זמנית במקום עבור כל חוט בנפרד.

g. מספר החוטים בהם בחרנו להשתמש הוא 1024. בחרנו במספר זה כיוון והוא מהווה כפולה שלמה של רוחב $TILE$ בודד (64) לכן עומד בדרישת התרגיל, הוא כפולה נוחה של גודל $TILE$ בודד ($64 \cdot 64 = 4096$) ולכן מאפשר חלוקה אחידה של חישוב היסטוגרמה בין כל החוטים (4 חישובים לחוט), ולא חורג מהגבלת מספר החוטים ל-$ThreadBlock$ יחיד במעבד הנתון.
בזמן איתחול ההיסטוגרמה, חישוב ה-CDF בעזרת prefix_sum, וחישוב ה-map לכל $TILE$, נדרש רק ל-256 חוטים, לכן בזמן זה יתר החוטים לא יבצעו חישוב.

h. זמני הריצה של התכנית הם:
נבחין כי זמן הריצה עבור המימוש הסריאלי הוא - $329.1\ [msec]$.
בזמן החישוב התכנית מעבדת 1000 תמונות, ולכן התפוקה היא – 3039 תמונות לשניה.

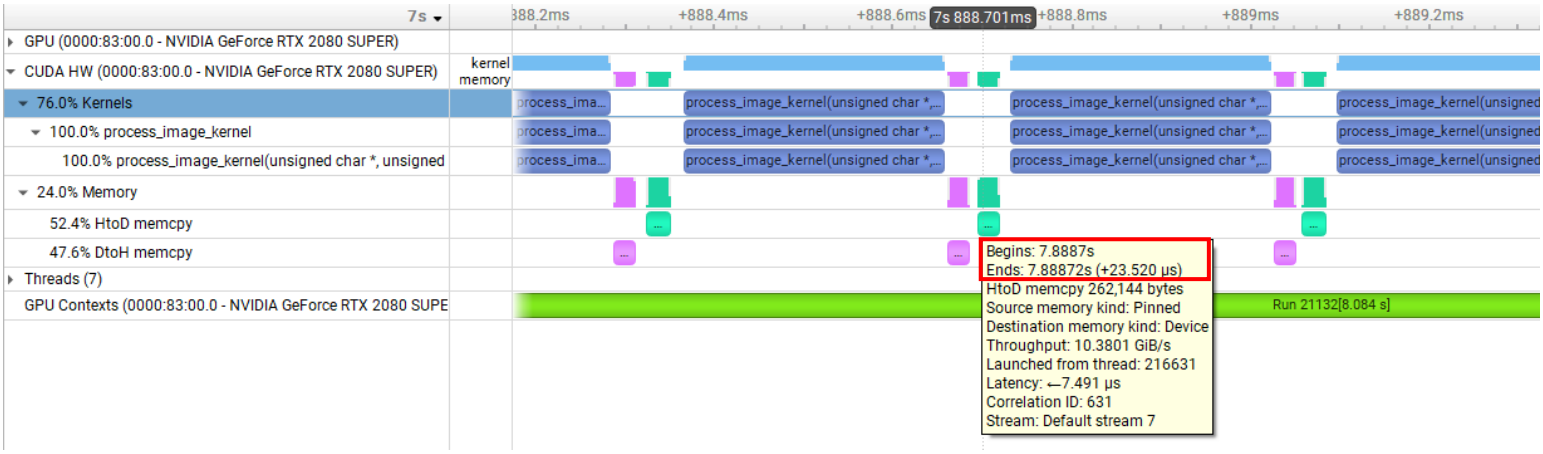
```
$ ./ex1
Number of devices: 4
Using device 2

=== Randomizing images ===
total time 1642.436378 [msec]

=== CPU ===
total time 4936.146139 [msec]

=== GPU Task Serial ===
total time 329.087564 [msec]  distance from baseline 0 (should be zero)

=== GPU Bulk ===
total time 51.209394 [msec]  distance from baseline 0 (should be zero)
```

i. להלן צילום מסך מתוך הרצת התכנית ב- $Nsight\ System$, בחלק בתכנית שבו רצים ה-$kernel$ בזה אחר זה עבור כל תמונה:



ניתן לראות קריאות ל-$kernel$ בכחול כהה, וכן את זמני העברת המידע בסגול מה-$device$ ל-$host$ ובירוק מה-$host$ ל-$device$.

j. נבחר למשל את אחד מההעתקות המידע מה-CPU ל-GPU. ניתן לראות שזמן ההעתקה הוא $23.52\ [\mu sec]$.



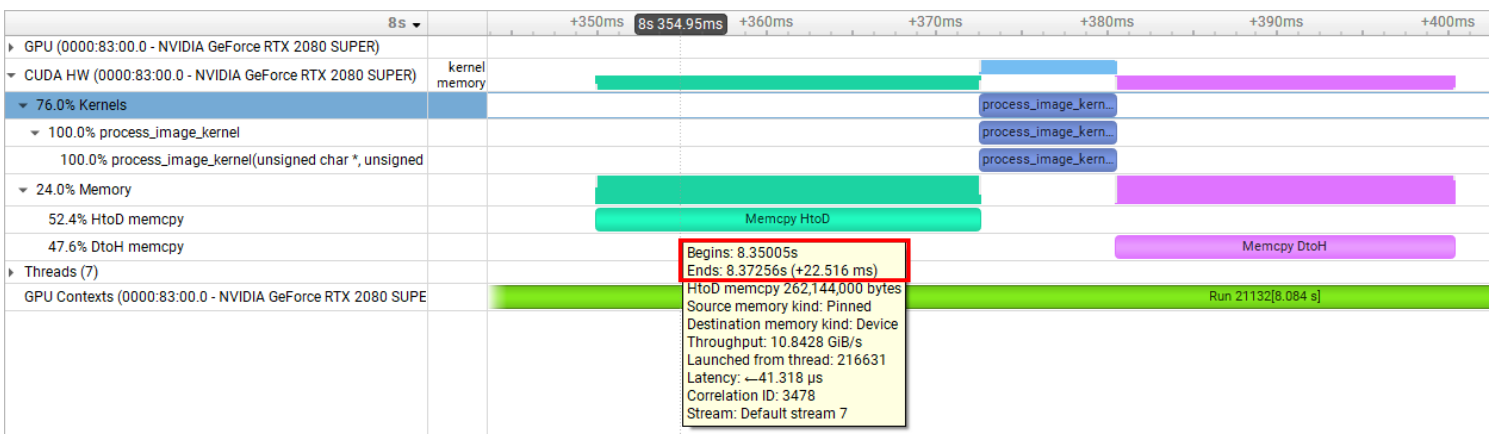
חלק 4

a. בחרנו לערוך את ה-$kernel$ הקודם כך שייתמוך בשני המימושים.

```
$ ./ex1
Number of devices: 4
Using device 2

=== Randomizing images ===
total time 1642.436378 [msec]

=== CPU ===
total time 4936.146139 [msec]

=== GPU Task Serial ===
total time 329.087564 [msec]  distance from baseline 0 (should be zero)

=== GPU Bulk ===
total time 51.209394 [msec]  distance from baseline 0 (should be zero)
```

f. זמני הריצה של התכנית הם:
נבחין כי זמן הריצה עבור המימוש $bulk$ הוא - $51.2\ [msec]$.
ביחס לזמן הריצה הסריאלי, נקבל $speedup = \frac{329.1}{51.2} = 6.43$.
מימוש זה מהיר יותר פי 6.4.

g. להלן צילום מסך מתוך הרצת התכנית ב- $Nsight\ System$, בחלק בתכנית שבו רץ $kernel$ אחד עם $ThreadBlocks$ כמספר התמונות הנתונות – 1000:



בצילום ניתן גם לראות מידע אודות העתקת הנתונים מה-CPU ל-GPU.

h. כפי שניתן לראות, בחלק זה של התכנית, זמן העתקה ל-GPU הוא $22.51\ [msec]$.
בהשוואה לסעיף $3.J$, ניתן לראות שזמן העתקה גדל ב-3 סדרי גודל (פי 1000),
כלומר לינארי בגודל המידע המועתק. (בסעיף $3.J$, הועתקה בכל פעם תמונה אחת בלבד, ובסעיף זה מועתקות כל 1000 התמונות באותו זמן).