**Department of Computer Engineering**



**Cairo University**

**Faculty of Engineering**

**Parallel Computing**

**Lab 2**

**Submitted to**

Eng. Mohamed Abdallah

**Submitted by**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Sec** | **BN** | **Code** |
| **Ahmed Sayed Sayed Madbouly** | **1** | **3** | **9202111** |
| **Ahmed Hany Farouk Tawfik** | **1** | **10** | **9202213** |

**Trials made.**

For matrix 100 x 300

Kernel 1:

A screenshot of a computer

Description automatically generated

Kernel 2:

A screenshot of a computer

Description automatically generated

Kernel 3:

A screenshot of a computer screen

Description automatically generated

For 300 \* 300

Kernel 1:

A screenshot of a computer

Description automatically generated

Kernel 2:

A screenshot of a computer

Description automatically generated

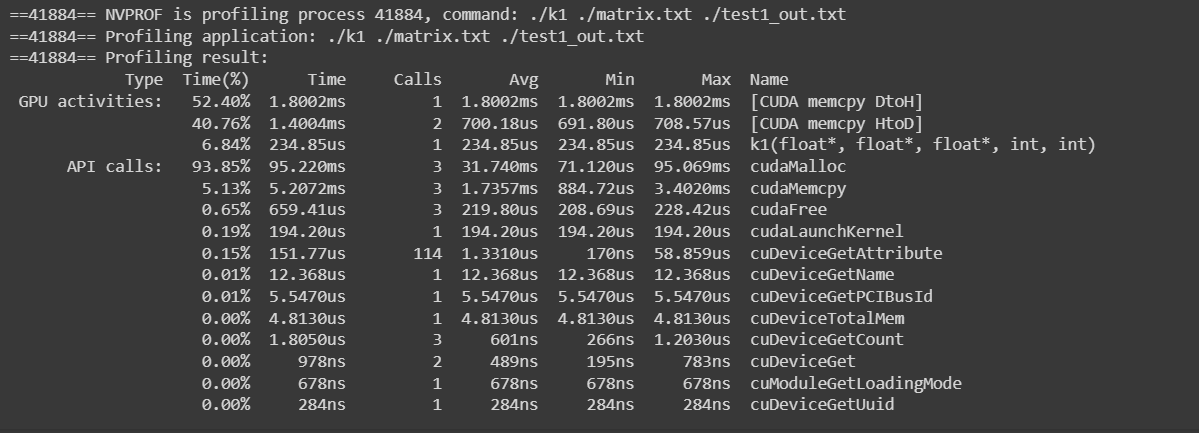
Kernel 3:

A screenshot of a computer screen

Description automatically generated

For 1000 \* 1000

Kernel 1:



Kernel 2:

A screenshot of a computer

Description automatically generated

Kernel 3:

A screenshot of a computer

Description automatically generated

For 10000\*100

Kernel 1:

A screenshot of a computer

Description automatically generated

Kernel 2:

A screenshot of a computer screen

Description automatically generated

Kernel 3:A screenshot of a computer program

Description automatically generated

For 100 \* 10000

Kernel 1:

A screen shot of a computer

Description automatically generated

Kernel 2:

A black screen with white text

Description automatically generated

Kernel 3:A screenshot of a computer screen

Description automatically generated

For 3 test cases 30\*50

Kernel 1:

A screen shot of a computer

Description automatically generated

Kernel 2:A screen shot of a computer

Description automatically generated

Kernel 3:A screen shot of a computer

Description automatically generated

**Conclusion**

**For test case 1: 100 \* 300**

we had number of rows < number of columns.

1. K1 will be faster.
2. K3 faster than k2 because number threads are larger in k3, so the threads compute fewer numbers than in k2.

**For test case 2: 300 \* 300:**

we had number of rows = number of columns.

1. K1 will be faster.
2. K3 is faster as it moves by rows to calculate column which may be faster them k2 which moves by columns to calculate the row.

**For test case 3: 1000 \* 1000:**

we had number of rows = number of columns.

1. K1 will be faster.
2. K3 is faster as it moves by rows to calculate column which may be faster them k2 which moves by columns to calculate the row.

**For test case 4: 10000\*100:**

we had number of rows > number of columns.

1. K1 will be faster.
2. K2 faster than k3 because number threads are larger in k2, so the threads compute fewer numbers than in k3.

**For test case 5: 100 \* 10000:**

we had number of rows < number of columns.

1. K1 will be faster.
2. K3 faster than k2 because number threads are larger in k3, so the threads compute fewer numbers than in k2.

**For test case 6: 3 test cases 30 \* 50:**

we had number of rows < number of columns.

1. K1 will be faster.
2. K3 faster than k2 because number threads are larger in k3, so the threads compute fewer numbers than in k2.