

## **ΑΝΑΦΟΡΑ ΕΡΓΑΣΙΑΣ 2 ΠΑΡΑΛΛΗΛΑ ΣΥΣΤΗΜΑΤΑ**

ΜΕΛΗ ΕΡΓΑΣΙΑΣ: ΠΑΡΑΣΚΕΥΑΣ ΤΣΙΛΗΣ ΑΜ: 1115202200195

ΜΠΑΛΩΜΕΝΟΣ ΙΑΣΟΝΑΣ ΑΜ: 1115202200104

### **ΑΣΚΗΣΗ 1:**

ΤΟ ΠΡΟΒΛΗΜΑ ΉΤΑΝ Ο ΠΟΛΛΑΠΛΑΣΙΑΣΜΟΣ ΔΥΟ ΠΟΛΥΩΝΥΜΩΝ Ν ΣΥΝΤΕΛΕΣΤΩΝ ΜΕ ΤΟΝ ΣΕΙΡΙΑΚΟ ΑΛΓΟΡΙΘΜΟ (2 FOR LOOPS) ΚΑΙ ΤΟΝ ΠΑΡΑΛΛΗΛΟ ΚΑΙ ΣΥΓΚΡΙΣΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ. ΕΠΙΣΗΣ ΕΠΡΕΠΕ ΝΑ ΓΙΝΕΙ Η ΟΡΘΟΤΗΤΑ ΤΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ. ΓΙΑ ΤΟΝ ΠΑΡΑΛΛΗΛΟ ΑΛΓΟΡΙΘΜΟ ΧΡΗΣΙΜΟΠΟΙΗΘΗΚΕ OPENMP ΟΠΟΥ ΚΑΘΕ THREAD ΠΑΙΡΝΕΙ ΕΝΑ ΚΟΜΜΑΤΙ ΑΠΟ ΤΟ ΕΞΩΤΕΡΙΚΟ FOR LOOP ΚΑΙ ΤΟ ΠΟΛΛΑΠΛΑΣΙΑΖΕΙ ΜΕ ΚΑΘΕ J ΑΠΟ ΤΟ INNER LOOP ΚΑΙ ΤΟ ΑΠΟΘΗΚΕΥΕΙ ΣΕ LOCAL BUFFER ΓΙΑ ΤΗΝ ΑΠΟΦΥΓΗ RACE CONDITIONS ΚΑΙ ATOMIC LOCKS ΚΕΡΔΙΖΟΝΤΑΣ ΧΡΟΝΟ. ΤΕΛΟΣ ΤΑ LOCAL BUFFERS ΣΥΓΧΩΝΕΥΟΝΤΑΙ ΚΑΙ ΓΙΝΕΤΑΙ Ο ΕΛΕΓΧΟΣ ΓΙΑ ΤΗΝ ΟΡΘΟΤΗΤΑ ΤΟΥ ΠΑΡΑΛΛΗΛΟΥ ΑΛΓΟΡΙΘΜΟΥ ΜΕ ΤΟΝ ΣΕΙΡΙΑΚΟ.

ΠΑΡΑΚΑΤΩ ΕΙΝΑΙ ΤΑ ΑΠΟΤΕΛΕΣΜΑΤΑ:

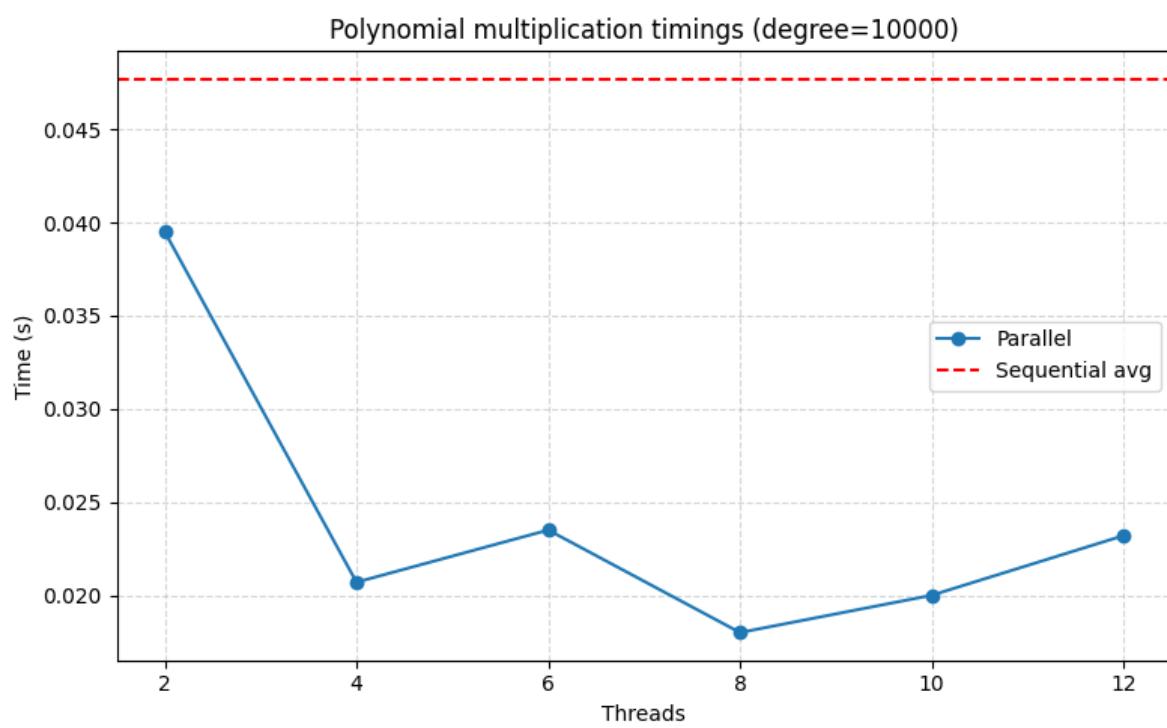
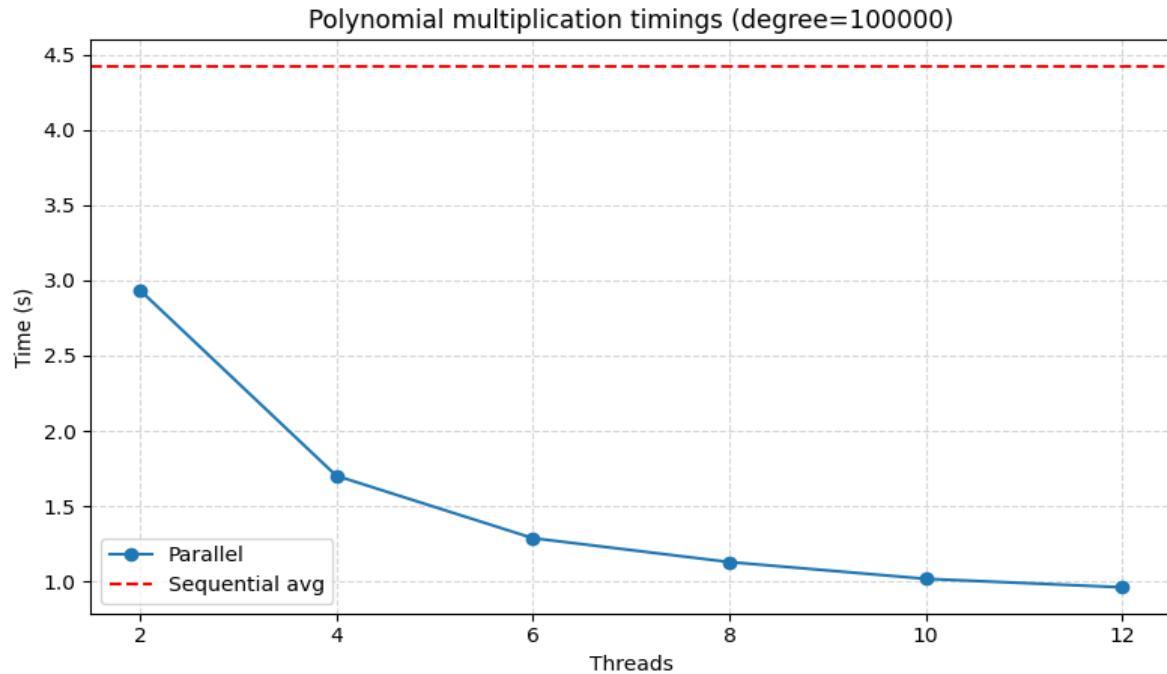
Parallel multiplication (OpenMP) — averages (s)

	2	4	6	8	10	12	Sequential
POLY. DEGREE: 10000	0.0395	0.0207	0.0235	0.0180	0.0200	0.0232	0.0477
POLY. DEGREE: 100000	2.9365	1.7022	1.2877	1.1290	1.0175	0.9612	4.4240

ΤΑ ΑΠΟΤΕΛΕΣΜΑΤΑ ΤΗ ΠΡΩΤΗΣ ΕΡΓΑΣΙΑΣ:

PARALLEL MULTIPLICATION	NUMBER OF THREADS						SEQUENTIAL
	2	4	6	8	10	12	
POLY. DEGREE: 10000	0,037	0,2	0,237	0,215	0,232	0,21	0,785
POLY. DEGREE: 100000	3,675	1,934	2,029	1,967	1,983	1,954	5,736

ΠΑΡΑΤΗΡΟΥΜΕ ΛΟΙΠΟΝ ΟΤΙ Η OPENMP ΕΙΝΑΙ ΕΛΑΦΡΩΣ ΓΡΗΓΟΡΟΤΕΡΗ ΣΤΟΝ ΠΑΡΑΛΛΗΛΟ ΑΛΓΟΡΙΘΜΟ. ΣΑΦΩΣ Η ΠΑΡΑΛΛΗΛΗ ΠΡΟΣΕΓΓΙΣΗ ΕΙΝΑΙ ΓΡΗΓΟΡΟΤΕΡΗ, ΠΑΡΑΤΗΡΕΙΤΑΙ ΕΠΙΣΗΣ ΟΤΙ Η ΔΙΑΦΟΡΑ ΣΤΗ ΧΡΗΣΗ ΤΟΝ ΑΡΙΘΜΟ ΤΩΝ THREADS (4 –12) ΕΙΝΑΙ ΑΠΕΙΡΟΕΛΑΧΙΣΤΗ. ΠΑΡΑΚΑΤΩ ΕΙΝΑΙ ΤΑ ΔΙΑΓΡΑΜΜΑΤΑ:



## ΑΣΚΗΣΗ 2:

ΣΕ ΑΥΤΗΝ ΤΗΝ ΑΣΚΗΣΗ ΣΚΟΠΟΣ ΉΤΑΝ ΝΑ ΦΤΙΑΞΟΥΜΕ ΕΝΑΝ ΤΥΧΑΙΟ ΠΙΝΑΚΑ (MxM) ΟΠΟΥ ΕΠΡΕΠΕ ΝΑ ΤΟΝ ΜΕΤΑΤΡΕΨΟΥΜΕ ΣΕ CSR ΠΑΡΑΛΛΗΛΑ ΚΑΙ ΕΠΕΙΤΑ ΘΑ ΠΟΛΛΑΠΛΑΣΙΑΖΕΙ ΠΑΡΑΛΛΗΛΑ ΤΟΝ CSR ΜΕ ΕΝΑ ΤΥΧΑΙΟ ΔΙΑΣΤΗΜΑ.

ΠΑΡΑΚΑΤΩ ΕΙΝΑΙ ΤΑ ΑΠΟΤΕΛΕΣΜΑΤΑ ΓΙΑ ITERATIONS = 20 ΔΙΑΦΟΡΕΤΙΚΑ SPARSITY ΚΑΙ ΜΕΓΕΘΟΣ 10^3 (ΟΛΑ ΤΑ ΑΠΟΤΕΛΕΣΜΑΤΑ ΒΡΙΣΚΟΝΤΑΙ ΑΝΑΛΥΤΙΚΑ ΣΤΑ PLOTS ΤΗΣ ΕΡΓΑΣΙΑΣ)

Exercise 2 summary table — m=10000, iterations=20 (times in seconds)

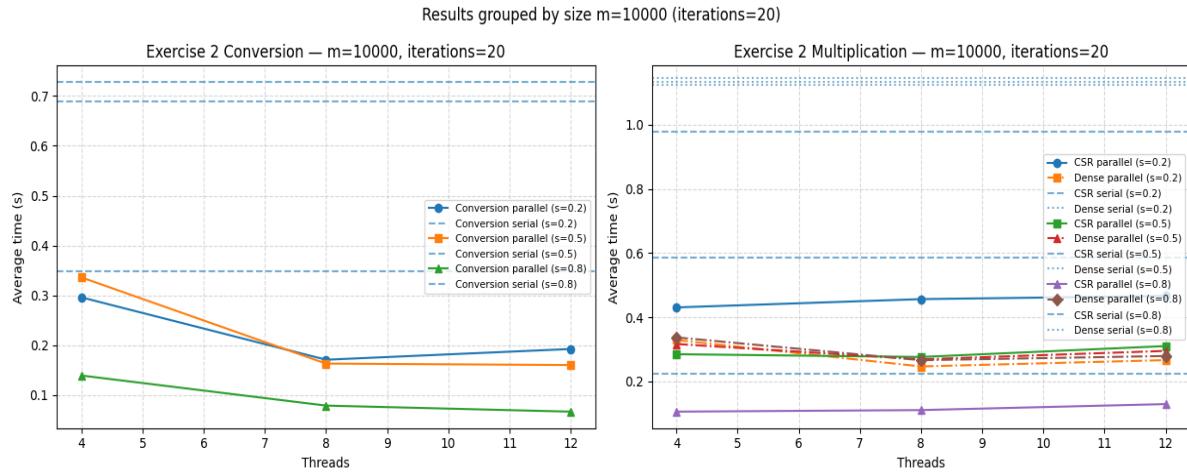
	4	8	12	Serial
Conversion parallel (s=0.2)	0.296058	0.170975	0.192463	-
Conversion serial (s=0.2)	-	-	-	0.727728
CSR parallel mult (s=0.2)	0.430715	0.456579	0.466239	-
CSR serial mult (s=0.2)	-	-	-	0.977965
Dense parallel mult (s=0.2)	0.329125	0.246912	0.266623	-
Dense serial mult (s=0.2)	-	-	-	1.132303
Conversion parallel (s=0.5)	0.336074	0.163227	0.160295	-
Conversion serial (s=0.5)	-	-	-	0.689606
CSR parallel mult (s=0.5)	0.285037	0.276433	0.310395	-
CSR serial mult (s=0.5)	-	-	-	0.584683
Dense parallel mult (s=0.5)	0.316844	0.269082	0.295328	-
Dense serial mult (s=0.5)	-	-	-	1.123048
Conversion parallel (s=0.8)	0.139285	0.078960	0.066838	-
Conversion serial (s=0.8)	-	-	-	0.348579
CSR parallel mult (s=0.8)	0.106164	0.111047	0.129680	-
CSR serial mult (s=0.8)	-	-	-	0.224554
Dense parallel mult (s=0.8)	0.337377	0.266327	0.279466	-
Dense serial mult (s=0.8)	-	-	-	1.145965

ΠΑΡΑΤΗΡΕΙΤΑΙ ΟΤΙ Ο ΠΑΡΑΛΛΗΛΟΣ ΤΡΟΠΟΣ ΓΙΑ ΝΑ ΔΗΜΙΟΥΡΓΗΘΕΙ Ο CSR ΚΑΙ Ο ΠΟΛΛΑΠΛΑΣΙΑΣΜΟΣ ΤΟΥ CSR ΜΕ ΤΑ ΔΙΑΝΥΣΜΑ ΕΙΝΑΙ ΣΑΦΕΣΤΑΤΑ ΓΡΗΓΟΡΟΤΕΡΟΣ ΑΠΟ ΤΟΝ ΣΕΙΡΙΑΚΟ(Ο ΧΡΟΝΟΣ ΠΕΦΤΕΙ 50%-60%).

ΟΣΟΝ ΑΦΟΡΑ ΤΟΝ ΧΡΟΝΟ ΓΙΑ ΤΟΝ ΠΟΛΛΑΠΛΑΣΙΑΣΜΟ ΠΥΚΝΗΣ ΜΟΡΦΗΣ(DENSE) ΣΕΙΡΙΑΚΑ ΕΝΑΝΤΙ ΠΑΡΑΛΛΗΛΑ ΚΑΙ ΕΚΕΙ ΒΛΕΠΟΥΜΕ ΔΙΑΦΟΡΕΣ 60%-70% ΜΕΙΩΣΗ ΧΡΟΝΟΥ.

ΟΣΟΝ ΑΦΟΡΑ ΤΗ ΣΥΓΚΡΙΣΗ ΠΑΡΑΛΛΗΛΗΣ ΥΛΟΠΟΙΗΣΗΣ ΓΙΑ DENSE ΚΑΙ CSR ΕΚΕΙ ΠΑΡΑΤΗΡΟΥΜΕ ΜΙΚΡΟΤΕΡΕΣ ΔΙΑΦΟΡΕΣ ΟΙ ΟΠΟΙΕΣ ΟΜΩΣ ΑΥΞΑΝΟΝΤΑΙ ΟΤΑΝ ΤΟ SPARSITY ΑΝΕΒΑΙΝΕΙ (ΔΗΛΑΔΗ ΑΥΞΑΝΕΤΑΙ Ο ΑΡΙΘΜΟΣ ΤΩΝ ΜΗΔΕΝΙΚΩΝ ΚΑΙ ΟΙ ΠΡΑΞΕΙΣ ΓΙΝΟΝΤΑΙ ΠΙΟ ΓΡΗΓΟΡΑ -> ΚΑΛΥΤΕΡΗ ΕΚΜΕΤΑΛΛΕΥΣΗ ΤΟΥ CSR).

ΑΚΟΛΟΥΘΟΥΝ ΤΑ ΔΙΑΓΡΑΜΜΑΤΑ:



### **ΑΣΚΗΣΗ 3:**

ΤΟ ΠΡΟΒΛΗΜΑ ΤΗΣ ΑΣΚΗΣΗΣ ΉΤΑΝ Η ΠΑΡΑΛΛΗΛΟΠΟΙΗΣΗ ΤΟΥ ΑΛΓΟΡΙΘΜΟΥ MERGESORT ΜΕ ΤΗΝ ΧΡΗΣΗ ΤΗΣ ΟΔΗΓΙΑΣ TASK ΤΗΣ OPENMP.

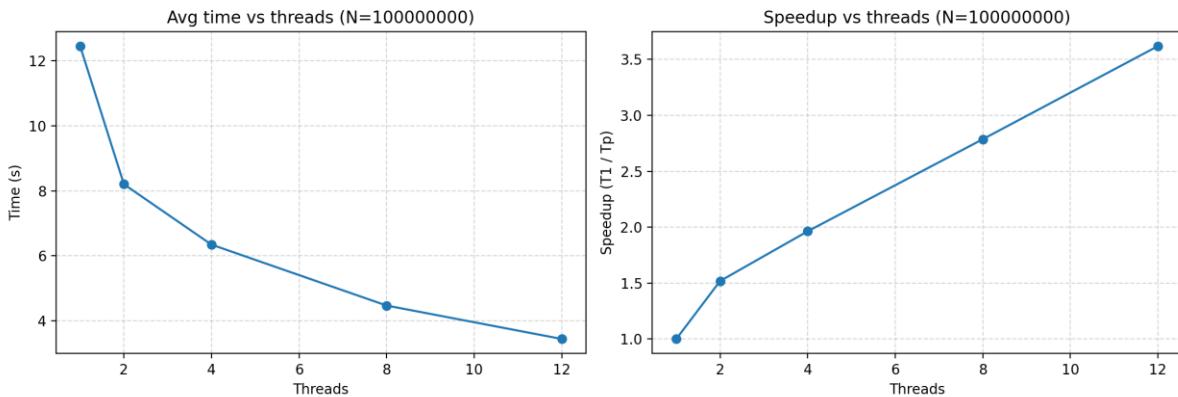
Exercise 3 — average time (seconds) (Serial = 1 thread)

	Serial	2	4	8	12
N=100000	0.009	0.006	0.005	0.005	0.011
N=10000000	1.087	0.715	0.590	0.374	0.317
N=100000000	12.452	8.202	6.344	4.469	3.441

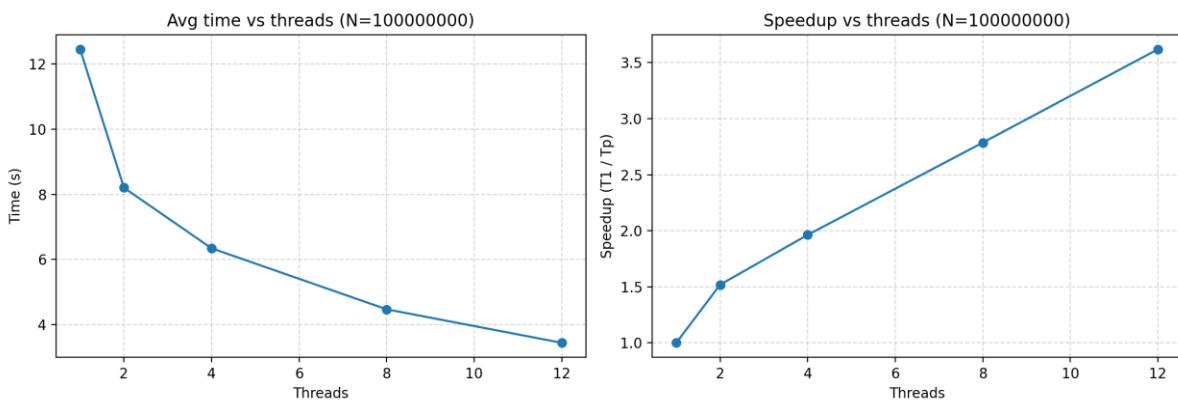
ΕΠΕΙΤΑ ΑΠΟ ΠΕΙΡΑΜΑΤΙΣΜΟ ΕΠΙΛΕΧΘΗΚΕ ΩΣ THRESHOLD ΓΙΑ ΤΗΝ ΔΗΜΙΟΥΡΓΙΑ ΝΕΟΥ TASK ΤΟ ΜΕΓΕΘΟΣ ΠΙΝΑΚΑ 10.000. ΑΝ ΤΟ ΜΕΓΕΘΟΣ ΤΟΥ ΠΙΝΑΚΑ ΣΤΗΝ ΚΛΗΣΗ ΕΙΝΑΙ < 10.000 ΕΠΙΛΕΓΕΤΑΙ Η ΧΡΗΣΗ ΤΟΥ ΣΕΙΡΙΑΚΟΥ ΑΛΓΟΡΙΘΜΟΥ, ΚΑΘΩΣ ΤΟ ΚΟΣΤΟΣ ΔΗΜΙΟΥΡΓΙΑΣ ΚΑΙ ΔΙΑΧΕΙΡΙΣΗΣ ΤΩΝ TASKS ΥΠΕΡΒΑΙΝΕΙ ΤΟ ΟΦΕΛΟΣ ΤΗΣ ΠΑΡΑΛΛΗΛΟΠΟΙΗΣΗΣ.

ΣΕ ΜΕΓΑΛΑ ΜΕΓΕΘΗ ΠΙΝΑΚΑ (10<sup>7</sup> ή 10<sup>8</sup>) ΠΑΡΑΤΗΡΕΙΤΑΙ ΣΗΜΑΝΤΙΚΗ ΜΕΙΩΣΗ ΤΟΥ ΧΡΟΝΟΥ ΕΚΤΕΛΕΣΗΣ. ΣΥΓΚΕΚΡΙΜΕΝΑ Η ΠΑΡΑΛΛΗΛΗ ΕΚΤΕΛΕΣΗ ΜΕ 16 THREAD ΕΜΦΑΝΙΖΕΙ ΜΕΙΩΣΗ ΕΩΣ ΚΑΙ 75% ΣΥΓΚΡΙΤΙΚΑ ΜΕ ΤΗΝ ΣΕΙΡΙΑΚΗ ΕΚΤΕΛΕΣΗ.

### Exercise 3 — mergesort



### Exercise 3 — mergesort



ΤΑ ΣΥΣΤΗΜΑΤΑ ΣΤΑ ΟΠΟΙΟ ΕΚΤΕΛΕΣΤΗΚΑΝ ΤΑ ΠΕΙΡΑΜΑΤΑ:

(DESKTOP-OQLOAOM)

Architecture: x86\_64 CPU op-mode(s): 32-bit, 64-bit Address sizes: 48 bits physical, 48 bits virtual Byte Order: Little Endian CPU(s): 12 On-line CPU(s) list: 0-11 Vendor ID: AuthenticAMD Model name: AMD Ryzen 5 3600 6-Core Processor CPU family: 23 Model: 113 Thread(s) per core: 2 Core(s) per socket: 6 Socket(s): 1 Stepping: 0 BogoMIPS: 7186.51 Flags: fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr sse sse2 ht syscall nx mmxext fxsr\_opt pdpe1gb rdtscp lm rep\_good nopl cpuid extd\_apicid tsc\_known\_freq pni pclm ulqdq ssse3 fma cx16 sse4\_1 sse4\_2 movbe popcnt aes xsave avx f16c rdrand hypervisor lahf\_lm cmp\_legacy cr8\_legacy abm sse4a misalignsse 3dnowprefetch osvw topoext ssbd ibpb stibp vmmcall fsgsbase bmi1 avx2 smep bmi2 rdseed adx smap clflushopt clwb sha\_ni xsaveopt xsavec xgetbv1 clzero xsaveerptr virt\_ssbd arat umip rdpid

ΛΕΙΤΟΥΡΓΙΚΟ ΣΥΣΤΗΜΑ: Linux DESKTOP-OQLOAOM 6.6.87.2-microsoft-standard-WSL2 #1 SMP PREEMPT\_DYNAMIC Thu Jun 5 18:30:46 UTC 2025 x86\_64 x86\_64 x86\_64 GNU/Linux

ΜΕΤΑΓΛΩΤΙΣΤΗΣ: gcc (Ubuntu 11.4.0-2ubuntu1~20.04) 11.4.0

LINUX10(ΣΧΟΛΗΣ):

Architecture: x86\_64 CPU op-mode(s): 32-bit, 64-bitByte Order: Little Endian  
Address sizes: 39 bits physical, 48 bits virtualCPU(s): 4On-line CPU(s) list: 0-3Thread(s)  
per core: 1Core(s) per socket: 4Socket(s): 1NUMA node(s): 1Vendor ID:  
GenuineIntelCPU family: 6 Model: 94Model name: Intel(R) Core(TM) i5-6500 CPU @  
3.20GHzStepping: 3CPU MHz: 3393.126CPU max MHz: 3600,0000CPU min MHz:  
800,0000

ΟΛΑ ΤΑ ΑΠΟΤΕΛΕΣΜΑΤΑ/ΓΡΑΦΗΜΑΤΑ ΕΙΝΑΙ ΑΝΑΛΥΤΙΚΑ ΣΤΟΥΣ ΦΑΚΕΛΟΥΣ ΤΗΣ  
ΕΡΓΑΣΙΑΣ RESULTS/PLOTS ΑΝΤΙΣΤΟΙΧΑ