



ΙΟΝΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΠΑΡΑΛΛΗΛΟΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ 2018

ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΗ ΕΡΓΑΣΙΑ #1

ΦΟΙΤΗΤΗΣ:
ΣΠΥΡΙΔΩΝ ΓΚΕΟΡΓΚ ΚΟΥΣΤΑΣ (Π2014027)

Οι δύο παραλλαγές του κώδικα υλοποιήθηκαν στα αρχεία matrix1.c και matrix2.c. Η διαφορά τους είναι ότι το ένα αρχείο προσπελαύνει τον πίνακα κατά σειρά και το άλλο κατα στήλη. Συγκεκριμένα, ο κώδικας διαφέρει στα εξής σημεία:

matrix1.c

```
for ( i = 0; i < numRows; i++){
    for (j = 0; j < numCol; j++) {
        //Prospellash ana grammh
        table[i*numCol+j]++;
    }
}
```

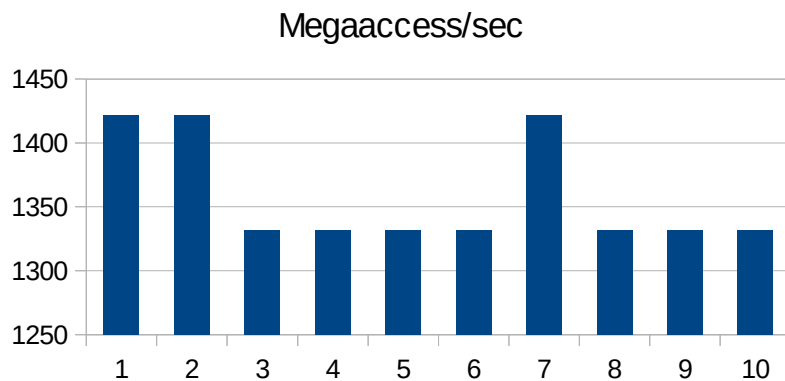
matrix2.c

```
for ( i = 0; i < numCol; i++){
    for (j = 0; j < numRows; j++) {
        //Prospellash ana sthlh
        table[i+j*numCol]++;
    }
}
```

Στο υπόλοιπο μέρος, ο κώδικας γράφει τα αποτελέσματα σε ένα αρχείο για να μπορέσουμε να τα αποθηκεύσουμε και να βγάλουμε τα εξής συμπεράσματα.

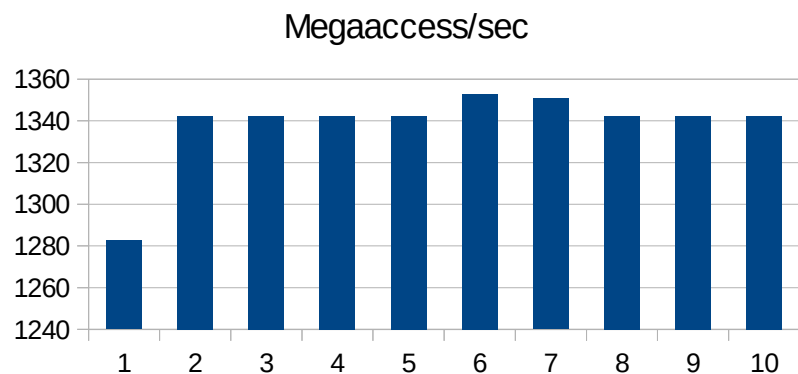
Όταν εκτελέστηκε ο κώδικας προσπέλλησης ανα σειρά είχαμε τα εξής αποτελέσματα:

NumRows = 100	
TIME	MEGA ACCESS/SEC
0.000014	1421.797966
0.000014	1421.797966
0.000015	1331.525079
0.000015	1331.525079
0.000015	1331.525079
0.000015	1331.525079
0.000015	1331.525079
0.000014	1421.797966
0.000015	1331.525079
0.000015	1331.525079
0.000015	1331.525079
0.0000147	1358.6069451



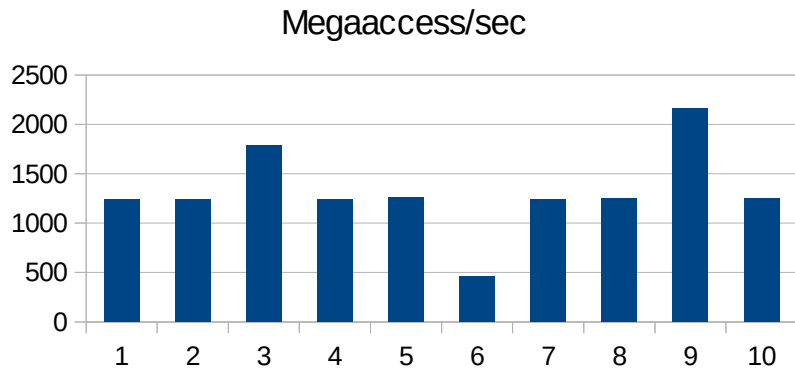
Για numRows=100 σειρές έχουμε μέσω χρόνο εκτέλεσης time= 0.0000147 και 1358.6069451 Megaaccess/sec. Γενικότερα η τελευταία σειρά περιγράφει τη μέση τιμή της κάθε στήλης.

NumRows = 1000	
TIME	MEGA ACCESS/SEC
0.000156	1282.661774
0.000149	1342.17728
0.000149	1342.17728
0.000149	1342.17728
0.000149	1342.17728
0.000149	1342.17728
0.000148	1353.00129
0.000148	1350.822544
0.000149	1342.17728
0.000149	1342.17728
0.000149	1342.17728
0.0001495	1338.1726568



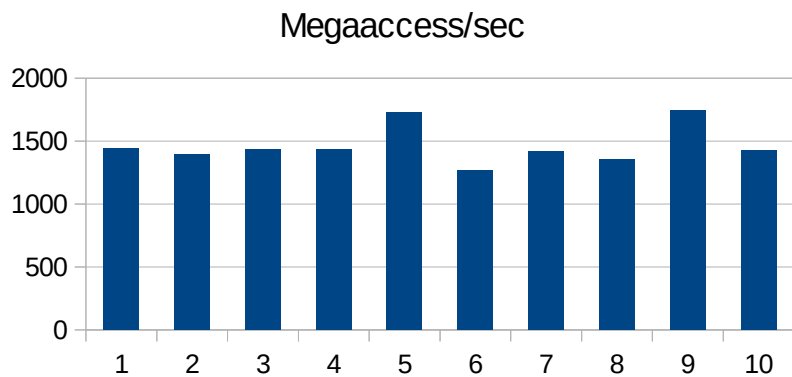
NumRows = 10000

TIME	MEGA ACCESS/SEC
0.001612	1240.734803
0.001608	1243.677984
0.00112	1785.57003
0.001621	1233.800265
0.001591	1257.096958
0.004309	464.151386
0.001609	1242.94088
0.001595	1253.902541
0.000925	2162.012371
0.001598	1251.470685
0.0017588	1313.5357903



NumRows = 100000

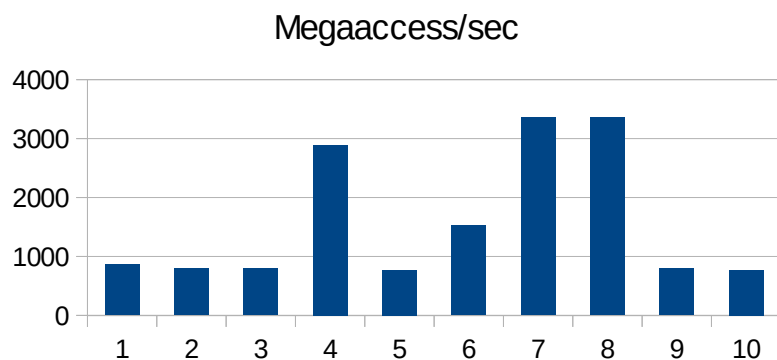
TIME	MEGA ACCESS/SEC
0.013861	1442.903487
0.014323	1396.355888
0.013973	1431.332093
0.013943	1434.391437
0.011583	1726.655003
0.015779	1267.505968
0.014126	1415.822714
0.014735	1357.314046
0.011462	1744.90026
0.014039	1424.598872
0.0137824	1464.1779768



Όταν εκτελέσαμε το πρόγραμμα για τον ίδιο αριθμό κόμβων (100, 1000, 10000 και 100000) αλλά προσπελλάσαμε ανα στήλη είχαμε τα εξής αποτελέσματα:

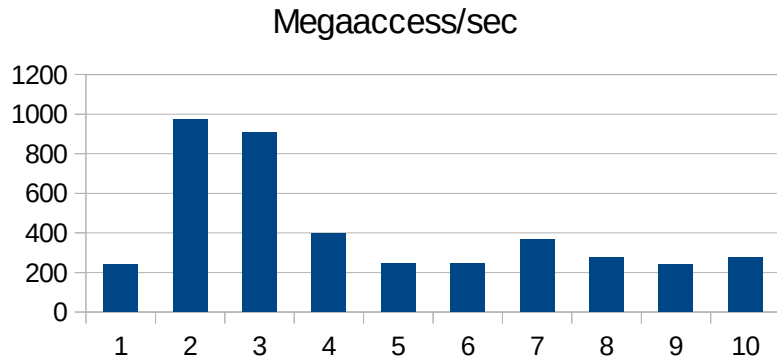
NumRows = 100

TIME	MEGA ACCESS/SEC
0.000023	864.804948
0.000025	798.915048
0.000025	798.915048
0.000007	2892.623448
0.000026	769.597064
0.000013	1525.201455
0.000006	3355.4432
0.000006	3355.4432
0.000025	798.915048
0.000026	769.597064
0.0000182	1592.9455523



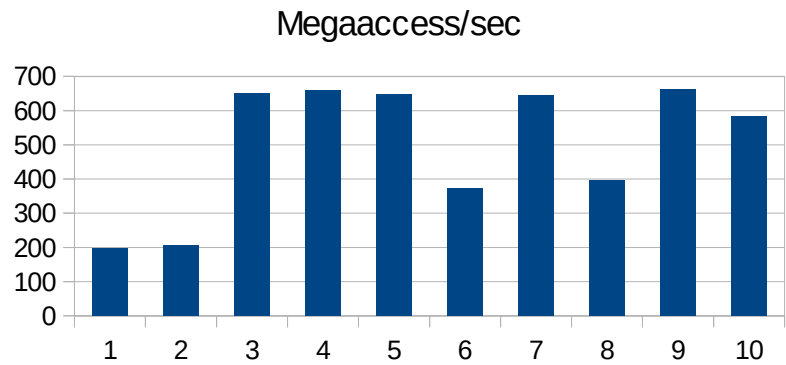
NumRows = 1000

TIME	MEGA ACCESS/SEC
0.000824	242.725926
0.000206	970.903704
0.000221	904.919957
0.000506	395.316117
0.000822	243.289095
0.000822	243.359675
0.000548	364.880731
0.00073	273.958459
0.000824	242.725926
0.00073	273.958459
0.0006233	415.6038049



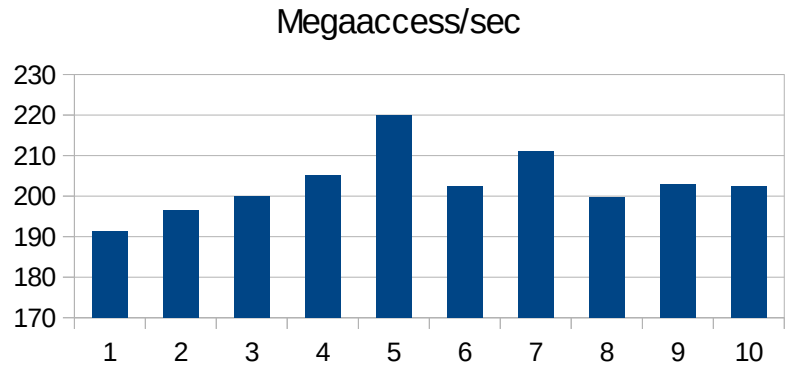
NumRows = 10000

TIME	MEGA ACCESS/SEC
0.010078	198.452993
0.009746	205.210822
0.003074	650.632746
0.003038	658.291454
0.003094	646.421207
0.005349	373.907199
0.003106	643.890697
0.005054	395.726389
0.003019	662.450288
0.003418	585.142857
0.0048976	502.0126652



NumRows = 100000

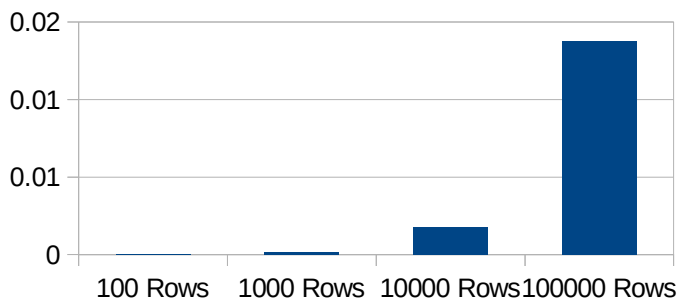
TIME	MEGA ACCESS/SEC
0.104508	191.372613
0.101786	196.490873
0.099973	200.054088
0.097508	205.111473
0.090965	219.865281
0.098825	202.377986
0.094739	211.106391
0.100098	199.803926
0.098618	202.803161
0.09873	202.572494
0.098575	203.1558286



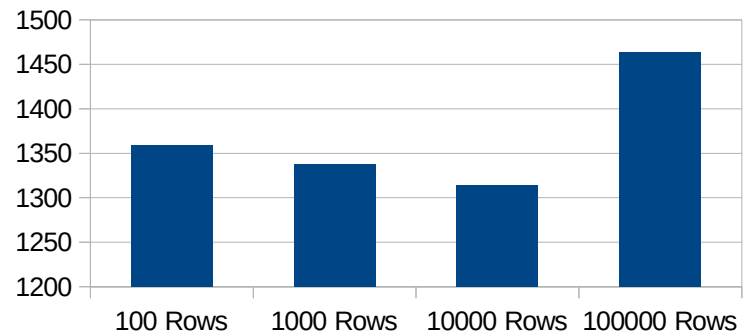
Ας συγκρίνουμε τώρα τους μέσους όρους χρόνου και megaaccess/sec όλων των παραδειγμάτων:

ΑΝΑ ΓΡΑΜΜΗ		
	Time	Megaaccess/sec
100 Rows	0.0000147	1358.6069451
1000 Rows	0.0001495	1338.1726568
10000 Rows	0.0017588	1313.5357903
100000 Rows	0.0137824	1464.1779768

Ανα Γραμμή - Time

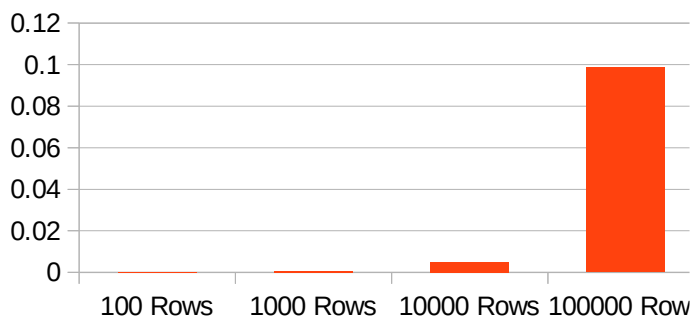


Ανα Γραμμή - Megaaccess/sec

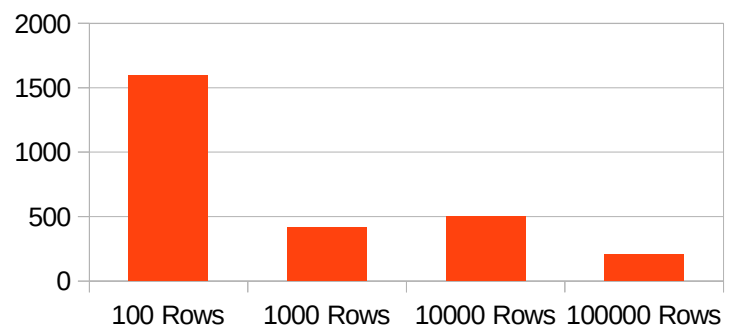


ΑΝΑ ΣΤΗΛΗ		
	Time	Megaaccess/sec
100 Rows	0.0000182	1592.9455523
1000 Rows	0.0006233	415.6038049
10000 Rows	0.0048976	502.0126652
100000 Rows	0.098575	203.1558286

Ανα Στήλη - Time



Ανα Στήλη - Megaaccess/sec



Παρατηρούμε ότι στο δεύτερο τρόπο προσπέλλαςης (ανα στήλη) ο αλγόριθμος εκτελέστηκε πιο αργά και με λιγότερα megaaccess/sec. Πράγμα που είναι και αναμενόμενο καθώς η σειριακή προσπέλλαση έχει μικρότερη πολυπλοκότητα.

Πηγές: Εργαστήριο του μαθήματος.