

## STATISTICS OF Multithreaded producer-consumer.

Όνοματεπώνυμο : Μιχάλης Καρατζάς

ΑΕΜ : 9137

email: [mikalaki@ece.auth.gr](mailto:mikalaki@ece.auth.gr)

Παρακάτω παρουσιάζονται και αναλύονται μερικά στατιστικά του χρόνου αναμονής μιας workFunction στην FIFO ουρά του προγράμματος, καθώς και τα συμπεράσματα που προκύπτουν από την ανάλυση. Τα στατιστικά υπολογίζονται για διάφορους συνδυασμούς των  $p$  και  $q$ , LOOP=10000, QUEUESIZE=10 και 1000. Τα στατιστικά αυτά προκύπτουν, από τα δεδομένα των csv που παράγονται από το πρόγραμμα για κάθε εκτέλεσή του και την διαχείριση αυτών των δεδομένων από το πρόγραμμα [statistics.m](#).

### • Ολικός μέσος χρόνος αναμονής

Queue size =10

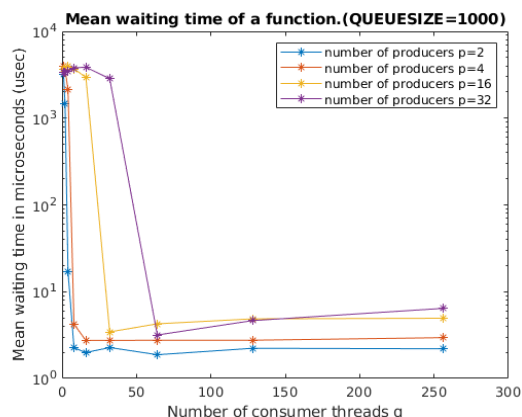
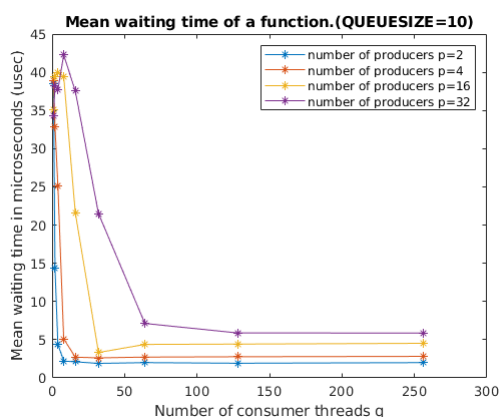
$p \setminus q$	q=1	q=2	q=4	q=8	q=16	q=32	q=64	q=128	Q=256
p=1	7.907	6.35	2.498	1.973	1.799	1.943	1.771	1.848	1.983
p=2	38.515	14.399	4.401	2.206	2.107	1.899	2	1.92	1.999
p=4	38.902	32.924	25.158	5.104	2.715	2.621	2.723	2.768	2.816
p=8	36.647	39.232	41.225	30.68	5.897	3.688	3.715	3.647	3.643
p=16	35.066	39.449	40.005	39.42	21.619	3.326	4.382	4.426	4.526
p=32	34.347	38.305	37.723	42.26	37.552	21.514	7.128	5.886	5.872
p=64	33.96	32.923	36.79	41.615	37.223	26.883	21.195	13.339	13.585

Πίνακας 1:Ολικός μέσος χρόνος αναμονής για QUEUESIZE=10

Queue size =1000

$p \setminus q$	q=1	q=2	q=4	q=8	q=16	q=32	q=64	q=128	Q=256
p=1	107.228	682.416	4.572	2.115	1.959	1.93	1.903	1.988	2.034
p=2	3230.505	1464.026	16.89	2.253	1.989	2.272	1.877	2.226	2.198
p=4	3997.201	3654.78	2120.548	4.224	2.741	2.748	2.754	2.758	2.952
p=8	3909.871	4726.561	4603.474	3086.12	3.148	3.254	3.6	3.829	3.659
p=16	3604.914	3677.429	4003.022	3688.1	2958.501	3.439	4.262	4.861	4.941
p=32	3374.09	3394.415	3493.08	3803.286	3858.888	2856.644	3.17	4.641	6.421
p=64	3349.608	3373.243	3679.007	4137.892	3930.922	3526.686	1846.586	3.548	7.643

Πίνακας 2:Ολικός μέσος χρόνος αναμονής για QUEUESIZE=1000



Από τις παραπάνω μετρήσεις, όπως αναφέρθηκε και στην αναφορά, βλέπουμε ότι κατά κανόνα καθώς αυξάνεται ο αριθμός των consumers  $q$  για σταθερό αριθμό producers  $p$ , ο μέσος χρόνος αναμονής μειώνεται έως ότου λάβει μια πολύ μικρή τιμή και έπειτα καθώς αυξάνουμε επιπλέον το  $q$ , βλέπουμε μικρές αυξομειώσεις (πρακτικά παραμένει σταθερός), κάτι το οποίο είναι εμφανές και στα παραπάνω διαγράμματα. Οπότε για συγκεκριμένο αριθμό παραγωγών  $p$ , έχουμε ελαχιστοποίηση του μέσου χρόνου αναμονής για συγκεκριμένο αριθμό καταναλωτών  $q$  (με πράσινο στους πίνακες) και η επιπλέον αύξηση του αριθμού των καταναλωτών μετά από αυτό το σημείο δεν μας δίνει πρακτικά κάποιο όφελος.

### • Μέσος χρόνος αναμονής για τα 3/4 των δεδομένων των χρόνων αναμονής

Παρακάτω παρουσιάζεται ο μέσος χρόνος αναμονής για τα 3/4 των δεδομένων των χρόνων αναμονής για διάφορους συνδυασμούς  $p$  και  $q$ . Μας ενδιαφέρει οι τιμές του να είναι παραπλήσιες με αυτές του συνολικού μέσου όρου, καθώς έτσι διαπιστώνεται σταθεροποίηση στις μετρήσεις μας για το 4ο τέταρτο των δεδομένων των χρόνων αναμονής που σημαίνει ότι τα συμπεράσματα που εξάγονται για τον χρόνο

αναμονής είναι ασφαλή , καθώς τα στατιστικά έχουν σταθεροποιηθεί. Παρακάτω παρουσιάζονται πίνακες μέσου χρόνου αναμονής για το 3/4 των δεδομένων:

### Queue size =10

p \ q	q=1	q=2	q=4	q=8	q=16	q=32	q=64	q=128	Q=256
p=1	7.644	6.693	2.504	1.995	1.816	1.95	1.748	1.902	2.009
p=2	41.32	14.352	4.584	2.268	2.159	1.936	2.066	1.95	2.052
p=4	40.116	34.501	25.681	4.663	2.76	2.585	2.773	2.846	2.964
p=8	36.847	40.025	42.107	32.263	5.9	3.899	3.863	3.739	3.748
p=16	35.396	39.851	40.063	40.768	22.438	3.383	4.51	4.448	4.537
p=32	34.083	38.646	39.104	42.338	37.141	26.508	7.062	6.848	5.722
p=64	34.04	33.626	37.689	42.127	37.654	27.429	20.998	12.156	14.014

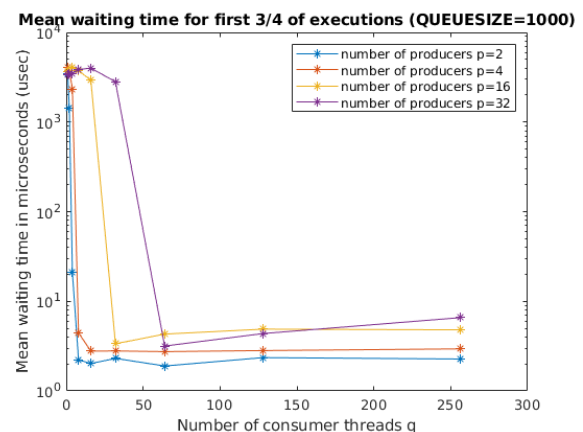
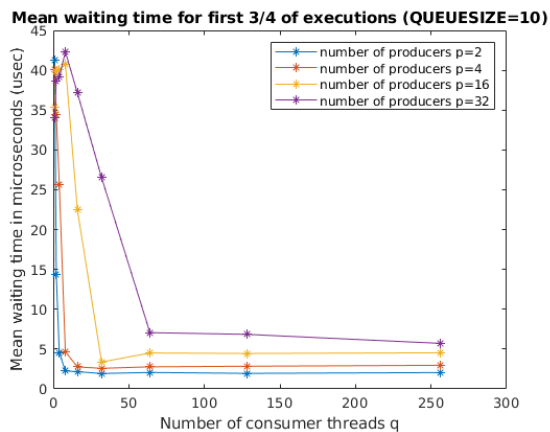
Πίνακας 3: Μέσος χρόνος αναμονής για τα 3/4 των δεδομένων των χρόνων αναμονής (QUEUE SIZE=10)

### Queue size =1000

p \ q	q=1	q=2	q=4	q=8	q=16	q=32	q=64	q=128	Q=256
p=1	129.265	620.704	4.499	2.134	2.023	1.959	1.956	2.034	2.129
p=2	3481.751	1413.774	20.845	2.221	2.02	2.306	1.897	2.352	2.269
p=4	4089.464	3762.176	2344.189	4.435	2.78	2.792	2.744	2.82	2.945
p=8	3880.412	4388.218	4702.959	3386.596	3.105	3.334	3.772	3.912	3.708
p=16	3604.905	3736.126	4101.541	3754.885	2922.81	3.359	4.307	4.889	4.803
p=32	3358.302	3429.288	3495.123	3859.064	3995.572	2813.277	3.167	4.365	6.564
p=64	3363.048	3343.893	3630.564	4149.279	3919.26	3557.495	1776.763	3.517	7.683

Πίνακας 4: Μέσος χρόνος αναμονής για τα 3/4 των δεδομένων των χρόνων αναμονής (QUEUE SIZE=1000)

Γραφήματα μέσου χρόνου αναμονής για τα 3/4 των εκτελέσεων των συναρτήσεων:



Από τους πίνακες παραπάνω(Πίνακες 3 και 4) βλέπουμε ότι πράγματι οι τιμές του μέσου χρόνου αναμονής για τα 3/4 των εκτελέσεων είναι πάρα πολύ κοντά στους ολικούς μέσους χρόνους αναμονής(Πίνακες 1 και 2), ενώ τα γραφήματα για τα 3/4 των δεδομένων παρουσιάζουν ίδια συμπεριφορά με αυτά του ολικού μέσου χρόνου αναμονής, παρακάτω φαίνονται και οι απόλυτες διαφορές μεταξύ των δύο στατιστικών :

### Queue size =10

p \ q	q=1	q=2	q=4	q=8	q=16	q=32	q=64	q=128	Q=256
p=1	0.262	0.343	0.005	0.022	0.017	0.007	0.023	0.054	0.026
p=2	2.805	0.047	0.183	0.061	0.053	0.037	0.066	0.03	0.053
p=4	1.214	1.578	0.523	0.441	0.045	0.037	0.05	0.078	0.148
p=8	0.2	0.793	0.882	1.583	0.003	0.212	0.147	0.092	0.105
p=16	0.33	0.402	0.059	1.347	0.819	0.056	0.127	0.022	0.011
p=32	0.264	0.341	1.381	0.078	0.411	4.994	0.065	0.962	0.151
p=64	0.08	0.703	0.9	0.512	0.431	0.546	0.197	1.183	0.429

Πίνακας 5: Απόλυτη διαφορά μεταξύ μέσου χρόνου αναμονής για 3/4 των δεδομένων και συνολικού μέσου χρόνου αναμονής (QUEUE SIZE=10)

### Queue size =1000

p \ q	q=1	q=2	q=4	q=8	q=16	q=32	q=64	q=128	Q=256
p=1	22.037	61.711	0.074	0.02	0.063	0.03	0.053	0.046	0.095
p=2	251.246	50.252	3.954	0.032	0.031	0.035	0.02	0.126	0.071
p=4	92.263	107.396	223.641	0.212	0.04	0.044	0.009	0.062	0.007
p=8	29.459	338.343	99.485	300.476	0.043	0.079	0.172	0.083	0.049
p=16	0.009	58.697	98.519	66.785	35.691	0.08	0.046	0.028	0.138
p=32	15.788	34.874	2.043	55.778	136.684	43.367	0.003	0.276	0.142
p=64	13.44	29.351	48.443	11.387	11.662	30.809	69.822	0.031	0.041

Πίνακας 6: Απόλυτη διαφορά μεταξύ μέσου χρόνου αναμονής για 3/4 των δεδομένων και συνολικού μέσου χρόνου αναμονής (QUEUE SIZE=1000)

Από τους πίνακες 5 και 6 σε συνδυασμό με τους πίνακες 1,3 και 2,4 , βλέπουμε ότι η διαφορά μεταξύ μέσου χρόνου για τα 3/4 των δεδομένων και του συνολικού μέσου χρόνου, είναι τάξεις μεγέθους μικρότερη από τις τιμές των στατιστικών αυτών και επομένως σε συνδυασμό με όσα διαπιστώθηκαν στην παραπάνω παράγραφο μπορούμε να συμπεράνουμε, ότι η τιμή LOOP=10000, είναι αρκετά μεγάλη ώστε να σταθεροποιηθούν τα στατιστικά μας και να λάβουμε ασφαλή συμπεράσματα για τους μέσους χρόνους αναμονής .

### • Τυπική απόκλιση χρόνων αναμονής

#### Queue size =10

p \ q	q=1	q=2	q=4	q=8	q=16	q=32	q=64	q=128	Q=256
p=1	8.64	6.565	7.963	4.266	1.952	2.918	1.354	1.407	3.181
p=2	21.177	15.392	5.987	13.042	12.241	1.426	10.028	4.71	9.91
p=4	69.074	39.517	43.018	31.541	14.811	12.655	20.905	20.73	24.211
p=8	59.862	69.675	84.851	58.086	18.558	41.877	38.41	35.512	36.118
p=16	66.097	106.547	102.671	109.36	57.675	25.144	51.558	50.392	53.813
p=32	61.972	123.786	114.908	141.734	128.617	33.898	21.26	49.964	68.938
p=64	61.037	88.535	125.789	156.498	124.545	37.774	26.458	25.569	35.214

Τυπική απόκλιση χρόνων αναμονής (QUEUE SIZE=10)

#### Queue size =1000

p \ q	q=1	q=2	q=4	q=8	q=16	q=32	q=64	q=128	Q=256
p=1	80.175	340.653	13.136	2.882	3.017	1.464	3.469	5.143	3.049
p=2	1770.506	539.343	37.896	11.214	3.739	19.19	6.901	24.965	11.297
p=4	782.155	871.73	1191.392	15.329	12.943	20.248	21.009	17.038	23.373
p=8	886.092	1975.131	1224.828	1374.703	13.084	30.606	36.114	40.376	34.042
p=16	619.073	702.983	1038.824	744.932	1479.741	29.803	46.214	59.416	56.967
p=32	413.523	683.517	644.403	1108.105	1115.186	1278.458	10.316	49.879	74.013
p=64	258.867	869.086	1184.714	1488.907	1374.579	968.736	1527.268	7.564	86.352

Τυπική απόκλιση χρόνων αναμονής (QUEUE SIZE=1000)

Όπως μπορούμε να δούμε η τυπική απόκλιση παίρνει τιμές αρκετά μεγάλες , που είναι συγκρίσιμες με τον μέσο χρόνο αναμονής , ενώ σε μερικές περιπτώσεις είναι και μεγαλύτερη! Μια σημαντική τυπική απόκλιση ήταν αναμενόμενη , καθώς κατά την διαδικασία εκτέλεσης του προγράμματος , θα υπάρξουν συναρτήσεις οι οποίες θα εκτελεστούν αμέσως μόλις προστεθούν στην ουρά αναμονής (κυρίως στην αρχή του προγράμματος , ενώ άλλες θα περιμένουν για πολύ χρόνο έως ότου έρθει η σειρά τους (ειδικά σε μεγάλες ουρές και όταν ο αριθμός των καταναλωτών είναι μικρός).

Βέβαια καθώς το **q** παίρνει μεγάλες τιμές για σταθερό **p** βλέπουμε ότι η τυπική απόκλιση αρχίζει και παίρνει μικρότερες τιμές (π.χ. για **p=16** και **q=32** σε σχέση με **p=16** και **q=4**). Αυτό σημαίνει όπως έχει αναφερθεί και στην αναφορά , ότι πλέον , οι consumers συνολικά "καταναλώνουν" και εκτελούν συναρτήσεις με μεγαλύτερο ρυθμό από ότι οι producers παράγουν με αποτέλεσμα , η ουρά να μην γεμίζει και γενικά οι χρόνοι αναμονής πλέον δεν παίρνουν μεγάλες τιμές και έτσι περιορίζεται η τυπική τους απόκλιση.

Από την άλλη πλευρά για μεγάλο αριθμό παραγωγών **p** , όταν ο αριθμός των consumers είναι πολύ μικρός (π.χ. **p=64** και **q=1**), έχουμε αρκετά μικρότερη τυπική απόκλιση από τι έχουμε για ελαφρώς μεγαλύτερο **q** (π.χ. **p=64** και **q=4** ή **q=8**), αυτό συμβαίνει γιατί σε αυτή την περίπτωση οι παραγωγοί παράγουν με πολύ γρηγορότερο ρυθμό από ότι καταναλώνουν οι consumers και έτσι λαμβάνουμε συνήθως μεγάλους χρόνους αναμονής, για το μεγαλύτερο μέρος της εκτέλεσης του προγράμματος , καθώς

η ουρά γεμίζει (εάν έχουμε φυσικά αρκετά μεγάλο αριθμό επαναλήψεων), με αποτέλεσμα να περιορίζεται η τυπική απόκλιση των χρόνων αναμονής.

(Τα παραπάνω συμπεράσματα γίνονται ευκολότερα αντιληπτά βλέποντας την ουρά με `QUEUE_SIZE=1000`, καθώς παίρνω μεγαλύτερα νούμερα τα οποία μπορώ ευκολότερα να συγκρίνω).

Συνοψίζοντας βλέπουμε ότι συναντάμε γενικά ,μικρότερες τιμές στην τυπική απόκλιση , όταν ο αριθμός των `consumer` είναι μεγάλος σε σχέση με τον αριθμό των `producers` και όταν αριθμός των `producers` είναι μεγάλος σε σχέση με τον αριθμό των `consumers`. Στην πρώτη περίπτωση οι συναρτήσεις καταναλώνονται αρκετά γρηγορότερα από ότι παράγονται ,με αποτέλεσμα ,μια μεγάλη ουρά να παραμένει πρακτικά άδεια , και οι χρόνοι αναμονής να περιορίζονται σε μικρές τιμές και στην δεύτερη , οι συναρτήσεις παράγονται με αρκετά μεγαλύτερο ρυθμό από ότι “καταναλώνονται” με αποτέλεσμα να γεμίζει η ουρά και να παίρνουμε κατά κανόνα μεγάλους χρόνους αναμονής.

**\*Όλα τα αρχεία και το MATLAB σκριπτάκι που οδήγησαν σε αυτή την ανάλυση υπάρχουν στον φάκελο [stats](#) \***