



## **Πρόβλημα Producers – Consumers Με πολλαπλούς Producers και έναν Consumer**

*της φοιτήτριας Ζαπαλίδη Ιωάννας  
1115201400044*



**Περιεχόμενα**

1. Δομή & Ανάπτυξη Κώδικα
2. Σχεδιαστικές Επιλογές
3. Μετρήσεις & Συμπεράσματα
4. Βιβλιογραφία

**Δομή & Ανάπτυξη Κώδικα**

Τα αρχεία της εργασίας είναι τα εξής:

- Semun.c & Semun.h – *συναρτήσεις σημαφόρων, από την προηγούμενη εργασία μου*
- shm.c & shm.h – *συναρτήσεις shared memory, από την προηγούμενη εργασία μου, αλλαγμένες για να υποστηρίζουν τα δεδομένα που χρειάζονται*
- main.c
- Makefile
- αυτό το αρχείο readme
- ένα randomfile.txt από όπου αντλούνται οι γραμμές με τυχαίο τρόπο

Εκτελούμε στο terminal, στο σωστό directory (όπου υπάρχουν και τα tracefiles):

- Για να γίνει compile: `$make`
- Για να γίνει εκτέλεση (με ενδεικτικές τιμές): `$make run`
- Για να γίνει εκτέλεση (με χρήση Valgrind): `$make val`
- Για καθαρισμό του directory: `$make clean`

**Σχεδιαστικές επιλογές:**

- Ο επιλεγμένος τρόπος IPC είναι, όπως και στην προηγούμενη εργασία, η χρήση shared memory, που συγχρονίζεται με χρήση σημαφόρων.
- Η διαμοιραζόμενη μνήμη in-ds έχει μέγεθος ενός struct που περιέχει ένα p\_id (του αποστολέα P) και ένα char[] που περιέχει το line που επιλέχθηκε τυχαία από το αρχείο.
- Η διαμοιραζόμενη μνήμη out-ds έχει μέγεθος ενός struct που περιέχει ένα p\_id (του αποστολέα P) και ένα int που περιέχει το hash.
- Η default τιμή για το πλήθος N των producers έχει οριστεί στο makefile, αλλά αλλάζει με επεξεργασία του makefile. Επιπλέον, αν δεν δωθεί από τον χρήστη έχει την τιμή που φαίνεται στο main.c
- Η default τιμή για το πλήθος K των επαναλήψεων έχει οριστεί στο makefile, αλλά αλλάζει με επεξεργασία του makefile. Επιπλέον, αν δεν δωθεί από τον χρήστη έχει την τιμή που φαίνεται στο main.c
- Η συνάρτηση md5 έχει εισαχθεί μέσω της βιβλιοθήκης openssl.
- Στη διαμοιραζόμενη μνήμη χωράει κάθε φορά ένα μήνυμα in και ένα out, συνεπώς δεν χρειάζεται να γίνει συγχρονισμός με mutexes, λαμβάνοντας υπόψιν ότι κάθε producer μπορεί να γράψει νέο msgin στην μνήμη μόνο εφόσον έχει παραλάβει κάποιο από την μνήμη out-ds και το έχει τυπώσει.
- Για την επικοινωνία των μετρητών από τις διεργασίες P προς την γονεϊκή χρησιμοποιήθηκαν οι συναρτήσεις return & wait, με τις P να επιστρέφουν τον μετρητή τους μέσω της return και την γονεϊκή να διαβάζει τις τιμές αυτές μέσω της wait. Αυτό φυσικά προσθέτει περιορισμούς στο μέγεθος των τιμών που μπορούν να επιστραφούν, αλλά επιλέχθηκε ως πιο απλός τρόπος για την

υλοποίηση της εργασίας.

### **Μετρήσεις & Συμπεράσματα**

Ακολουθούν ενδεικτικές τιμές κατά την εκτέλεση του προγράμματος:

<b>PNUM</b>	<b>K</b>	<b>SUM OF COUNTERS</b>
2	5	5
10	10	10
15	10	10
10	15	15
10	200	200
20	100	100
10	150	143
20	150	148
15	600	600

Παρατηρούμε ότι οι μετρητές εν γένει είναι υψηλοί σε σχέση με την τιμή K, καθώς ο συγχρονισμός των διεργασιών λόγω των ειδικών συνθηκών της εκφώνησης αποτελεί καλό περιβάλλον για να ταυτίζονται συχνά τα process ids. Πιο συγκεκριμένα, εφόσον κάθε φορά στην διαμοιραζόμενη μνήμη χωράνε το πολύ 1 μήνυμα από κάποια P στον C και 1 μήνυμα από τον C σε κάποιο P, και λαμβάνοντας υπόψιν ότι κάθε P για να στείλει νέο μήνυμα οφείλει να έχει πρώτα διαβάσει ένα άλλο, δεν αποτελεί έκπληξη ότι δεν υπάρχουν πολλές ενδείξεις διαφορετικών process ids.

### **Βιβλιογραφία**

- 1) Σημειώσεις μαθήματος
- 2) Αρχεία προηγούμενης εργασίας μου – shm.c, shm.h, Semun.c, Semun.h
- 3) MD5 man page