

# Λειτουργικά Συστήματα

## ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΉ ΆΣΚΗΣΗ 2

Σιρινιάν Αράμ Εμμανουήλ  
ΑΜ: 2537  
ae.sirinian@gmail.com

Απρίλιος 2019

### Ο στόχος

Στον κώδικα που παραδόθηκε έγινε μια προσπάθεια να επεκταθεί ένα λειτουργικό σύστημα. Σε ένα αντίγραφο του MINIX 3.2.0 να προστεθεί μια πολιτική δίκαιης χρονοδρομολόγησης.

### Τι έχει υλοποιηθεί

Αυτά που υλοποιήθηκαν είναι τα εξής:

- 1 Έχει υλοποιηθεί η μεταφορά της ταυτότητας του οδηγού ομάδας μιας διεργασίας `pm` στο `sched`. Αυτή η μεταφορά τελειώνει μέσα στην `schedule_process()` στο `servers/sched/schedule.c` μαζί με την προτεραιότητα της διεργασίας, το `κβάντο` και τους επεξεργαστές που μπορεί να εκτελεσθεί η διεργασία. Αφού έχει ξεκινήσει από την `do_fork()` του `servers/pm/forkexit.c` με την εισαγωγή του πεδίου `mp_procgrp` του `servers/pm/mproc.h` στο μήνυμα που στέλνει.
- 2 Έχει τροποποιηθεί η δομή `schedproc` στο `schedproc.h` για να συμπεριλαμβάνονται τα πεδία `procgrp`, `proc_usage`, `grp_usage` και `fss_priority`. Τα πεδία αυτά αρχικοποιούνται στην μέθοδο `do_start_scheduling()` στο `servers/sched/schedule.c`.
- 3 Όταν μια διεργασία ολοκληρώνει ένα `κβάντο`, ενημερώνονται τα πεδία `proc_usage`, `grp_usage` και `fss_priority` για όλες τις διεργασίες που περιμένουν στην ουρά χρήστη με την ίδια ταυτότητα οδηγού ομάδας με αυτήν που ενημερώνετε λόγω λήξης `κβάντου` διεργασίας. Αυτά στην μέθοδο `do_noquantum()` στο `servers/sched/schedule.c`.

- 4 Επίσης έχει τροποποιηθεί η δομή `proc` στο `proc.h` για να συμπεριλαμβάνει το πεδίο `p_fss_priority` με το οποίο θα πρέπει ο `kernel` να αποφασίζει μέσα σε ποια σημεία της δομής ουράς διεργασιών που έχει, να τοποθετήσει τις διεργασίες χρήστη.
- 5 Στο `src/include/minix/config.h` έχει μειωθεί το πλήθος ουρών έτσι ώστε οι διεργασίες χρήστη να εισάγονται μόνο σε μια ουρά (`USER_Q`). Επίσης ο `sched` ενημερώνει τον `kernel` με το `fss_priority`.
- 6 Τέλος επειδή ο διακομιστής διατρέχει περιοδικά τον πίνακα διεργασιών και τοποθετεί κάθε διεργασία σε ουρά υψηλότερης προτεραιότητας. Για αυτόν τον λόγο ο κώδικας της `balance_queues()` στο `servers/sched/schedule.c` δεν κάνει τίποτα σε περίπτωση που η ουρά είναι η `USER_Q`.

### Τι δεν έχει υλοποιηθεί

Αυτά που δεν έχουν υλοποιηθεί είναι τα εξής:

- 1 Πρώτον δεν ενημερώνονται τα `process_usage` και `group_usage` όταν τελειώνει το χβάντο μιας διεργασίας έχοντας ως αποτέλεσμα η ενημέρωση των πεδίων `process_usage`, `group_usage` και `fss_priority` να γίνεται με παλιότερες τιμές οι οποίες ασφαλώς είναι λανθασμένες.
- 2 Δεύτερον κατά την εκτέλεση της μεθόδου `switch_to_user()` στο `kernel/proc.c` καλούνται οι μέθοδοι `enqueue()` και `enqueue_head()`. Για να μπορεί να διαλέγει ο πυρήνας την διεργασία χρήστη με την χαμηλότερη τιμή αρκεί να αλλάξουμε τον τρόπο που εισάγονται τα στοιχεία στην ουρά, δηλαδή πρέπει να μπαίνουν ταξινομημένα, με το `head`, αυτό με το μικρότερο `fss_priority`.

### Τα αρχεία που τροποποιήθηκαν

Τα αρχεία που τροποποιήθηκαν είναι τα παρακάτω. Αντικαταστήσετε τα με αυτά που υπάρχουν στο πρωτότυπο αντίγραφο του MINIX 3.2.0. Μετά, αφού μείτε στον κατάλογο `/usr/src/` εκτελέστε τις εντολές `make libraries` και `make includes`. Συνεχίζοντας μείτε στον κατάλογο `/usr/src/tools/` και εκτελέστε `make install` για να χτίσετε το νέο σύστημα και να εγκαταστήσετε το νέο `image`.

- 1 `/usr/src/kernel/kernel.h`
- 2 `/usr/src/kernel/proto.h`
- 3 `/usr/src/kernel/system.c`
- 4 `/usr/src/kernel/proc.h`
- 5 `/usr/src/kernel/proc.c`

6 /usr/src/kernel/system/do\_schedule.c  
7 /usr/src/kernel/system/do\_schedctl.c  
8 /usr/src/include/minix/syslib.h  
9 /usr/src/include/minix/com.h  
10 /usr/src/include/minix/config.h  
11 /usr/src/include/minix/ipc.h  
12 /usr/src/servers/pm/mproc.h  
13 /usr/src/servers/pm/forkexit.c  
14 /usr/src/servers/sched/schedule.c  
15 /usr/src/servers/sched/schedproc.h  
16 /usr/src/lib/libsys/sys\_schedule.c

## Εξφαλμάτωση του κώδικα

Για τα κομμάτια που έχουν υλοποιηθεί έχουν γίνει διάφοροι έλεγχοι οι οποίοι όλοι πραγματοποιήθηκαν με την χρήση του `printf()`. Δυστυχώς τα νέα `images` που προκύπτουν από την μεταγλώττιση του κώδικα μετά την αντικατάσταση των αρχείων που έχουν τροποποιηθεί. Δεν τρέχουν σωστά αλλά κρασάρουν. Αυτό υποθέτω ότι συμβαίνει γιατί κάποιο αρχείο δεν συμπεριλαμβάνεται από αυτά που έχω τροποποιήσει και δυστυχώς λόγω της μεγάλης έκτασης του κώδικα είναι δύσκολο να βρεθεί. Αν είχα παραπάνω χρόνο θα ξεκινούσα να αντικαθιστώ κομμάτι κομμάτι και να τρέχω μέχρι να βρω τι χάθηκε αλλά για την ώρα αυτό είναι.

## Ομάδα

Στην ομάδα είναι μόνο ένα άτομο (Σιρινιάν Αράμ Εμμανουήλ, 2537)