

Εργασία 2

Σιάμπου Μαρία Δέσποινα
sdi1600151

Compilation:

Για να τρέξετε το πρόγραμμα εκτελέστε τις παρακάτω εντολές:

- make
- ./ex2 arg1 arg2 arg3 arg4 (arg5), όπου
 - arg1: ο αλγόριθμος αντικατάστασης (LRU ή WS)
 - arg2: ο συνολικός αριθμός πλαισίων (number of frames)
 - arg3: ο αριθμός εντολών q ανά διεργασία
 - arg4: ο μέγιστος αριθμός εντολών που θα διαβαστεί ανά διεργασία
 - arg5: το μέγεθος παραθύρου (αν έχουμε WS)

Περιληπτική Εξήγηση:

Αρχικά, το πρόγραμμα διαβάζει τα 2 αρχεία (που μας έχουν δοθεί) και τους αφαιρεί το offset, έτσι ώστε να αποκτήσουμε το page number (#p). Στη συνέχεια, το πρόγραμμα διαβάζει q “εντολές” εναλλάξ και από τα 2 αρχεία, έως ότου να διαβαστεί ο μέγιστος αριθμός “εντολών” που έχει δοθεί από τον χρήστη. Έπειτα, κάθε #p περνά στη MMU (Memory Management Unit). Η MMU, με τη σειρά της, εισάγει κάθε #p σε ένα Hashed Page Table και αντιστοιχίζεται σε ένα frame number. Πιο αναλυτικά:

- **Hashed Page Table:** Πρόκειται για έναν πίνακα λιστών. Κάθε entry της λίστας περιέχει ένα key (το page number δηλαδή), έναν αριθμό πλαισίου, ένα time signature το οποίο ανανεώνεται κάθε φορά που θέλουμε να εισάγουμε ένα page number που υπάρχει ήδη στο hash table, το RWbit που δίνεται από το αρχείο (R or W), ένα εικονικό pid για να διαχωρίζουμε τα #p που προέρχονται από διαφορετικές διεργασίες, και τέλος ένα δείκτη στον επόμενο κόμβο.
- **LRU:** Αν ο αλγόριθμος αντικατάστασης που επιλέξαμε είναι ο LRU τότε:
Ελέγχεται αν ο αριθμός σελίδας (#p) που θέλουμε να εισάγουμε στον πίνακα υπάρχει ήδη.
 - Αν δεν υπάρχει, δημιουργείται page fault και στη συνέχεια:
 - Αν η μνήμη δεν είναι γεμάτη, τότε το #p εισάγεται στον Page Table.
 - Αν η μνήμη είναι γεμάτη, διαγράφεται το #p με το μικρότερο time signature και στη συνέχεια εισάγεται το νέο #p.
 - Αν υπάρχει, τότε ανανεώνεται το time signature του και δεν ξαναεισάγεται στον Page Table.
- **WS:** Αν ο αλγόριθμος αντικατάστασης που επιλέξαμε είναι ο WS τότε:
Αρχικά ελέγχεται αν η μνήμη είναι γεμάτη.
 - Αν δεν είναι, τότε σε κάθε εισαγωγή ελέγχουμε αν το #p υπάρχει ήδη στο Page Table.

- Αν δεν υπάρχει, τότε το εισάγουμε και στη συνέχεια ενημερώνουμε το Working Set.
 - Αν το #p δεν υπάρχει στο Working Set, τότε:
 - Αν έχουν γίνει ήδη οι πρώτες n εισαγωγές (όπου n = μέγεθος παραθύρου) , δημιουργείται page fault. Στην συνέχεια, το #p με το μικρότερο time signature αντικαθιστάται από το νέο και έτσι ενημερώνεται το Working Set.
 - Αν δεν έχουν γίνει οι πρώτες n εισαγωγές, τότε το νέο #p εισάγεται απευθείας στο Working Set.
 - Αν το #p υπάρχει στο Working Set, τότε:
 - Αν έχουν γίνει ήδη οι πρώτες n εισαγωγές (όπου n = μέγεθος παραθύρου), διαγράφεται το #p με το μικρότερο time signature και στην συνέχεια αν το νέο #p εξακολουθεί να βρίσκεται στο Working Set, δεν ξαναισάγεται. Σε αντίθετη περίπτωση, το νέο #p εισάγεται στο Working Set.
 - Αν δεν έχουν γίνει οι πρώτες n εισαγωγές, τότε το νέο #p δεν ξαναισάγεται στο Working Set.
- Αν το #p υπάρχει στο Page Table, τότε απλά κάνουμε ενημέρωση του Working Set όπως περιγράφηκε παραπάνω.
 - Αν η μνήμη είναι γεμάτη, τότε διαγράφονται από τη μνήμη όλες οι σελίδες που δεν βρίσκονται στο Working Set.

Η μνήμη θεωρείται γεμάτη όταν ο αριθμός εισαγωγών είναι ίσος με τον συνολικό αριθμό πλαισίων της μνήμης.

Κάθε φορά που γίνεται εισαγωγή αριθμού σελίδας στο Page Table, αντιστοιχίζεται στον αριθμό αυτό και ένα frame από τη μνήμη. Η αντιστοίχιση αυτή γίνεται ως εξής: Έχουμε έναν πίνακα μεγέθους ίσου με τον συνολικό αριθμό πλαισίων. Ο πίνακας αρχικοποιείται με μηδενικές τιμές. Όταν μία νέα σελίδα εισάγεται στον Page Table τότε γίνεται αναζήτηση στον πίνακα πλαισίων. Ο αριθμός του πρώτου κελιού που θα έχει τη τιμή 0 θα δοθεί ως frame στον αντίστοιχο αριθμό σελίδας. Στη συνέχεια η τιμή του συγκεκριμένου κελιού αλλάζει σε 1.

Κάθε φορά που διαγράφεται ένα page number (#p) του οποίου το RWbit είναι W, τότε η αλλαγή θα πρέπει να γραφτεί στον δίσκο (DiskWrites++).