Η/Υ 445: Λειτουργικά Συστήματα
Άνοιξη 2002 - Παναγιώτα Φατούρου
3ο Σετ Προγραμματιστικών Ασκήσεων

Ημερομηνία Παράδοσης: Παρασκευή, 26/4/02, ώρα 18:00.

Σχόλια: Θα πρέπει να στείλετε με e-mail την άσκησή σας στη διεύθυνση cs445@cs.uoi.gr μέχρι τις 18:00 το απόγευμα της Παρασκευής, 26/4. Επίσης, θα πρέπει να κρατήσετε την άσκησή σας σε κάποιο directory στην περιοχή σας χωρίς να αλλάξετε την ημερομηνία της μέχρι να γίνει η εξέτασή της. Η ημερομηνία τελευταίας εγγραφής της άσκησής σας θα πρέπει να είναι είτε κάποια προηγούμενη ή η αναγραφόμενη ημερομηνία και ώρα παράδοσης. Μην ετοιμάζετε την άσκησή σας την τελευταία στιγμή. Η ημερομηνία και ώρα παράδοσης προβλέπει πιθανά προβλήματα στο σύστημα και η άσκησή σας θα πρέπει να παραδοθεί μέχρι αυτή την ημερομηνία ανεξάρτητα από τέτοιου είδους προβλήματα. Σας υπενθυμίζω πως ο κώδικάς σας θα πρέπει να είναι γραμμένος σε C, να τρέχει στα μηχανήματα του εργαστηρίου του Τμήματος, να είναι ευανάγνωστος και να έχει σχολιαστεί κατάλληλα.

Ο σκοπός της άσκησης είναι να εξοικειωθείτε με την έννοια και τις τεχνικές της ιδεατής μνήμης και της σελιδοποίησης, υλοποιώντας έναν προσομοιωτή ενός συστήματος διαχείρισης σελίδων.

Το πρόγραμμα σας θα πρέπει να δημιουργεί μία διεργασία διαχείρισης σελίδων (ΔΔΣ) και ένα σύνολο διεργασιών (ΔΠΔ) που επαναληπτικά ζητούν να προσβούν διευθύνσεις μνήμης, τις οποίες παράγουν με μια γεννήτρια τυχαίων αριθμών (οι διεργασίες θα δημιουργηθούν με τη γνωστή κλήση συστήματος fork() του UNIX). Οι διευθύνσεις αυτές θα πρέπει να διοχετεύονται στη διεργασία διαχείρισης σελίδων, η οποία υλοποιεί τον αλγόριθμο της λιγότερο-πρόσφατα χρησιμοποιούμενης σελίδας.

Περιγραφή Διεργασίας Διαχείρισης Σελίδων (ΔΔΣ)

Η ΔΔΣ εξυπηρετεί τις αιτήσεις πρόσβασης στη μνήμη που παράγουν οι ΔΠΔ. Η βασική δομή δεδομένων που πρέπει να διατηρεί η ΔΔΣ είναι ένας ανεστραμμένος πίνακας σελίδων. Υποθέστε ότι κάθε σελίδα αποτελείται από 1000 bytes, ενώ η κύρια μνήμη αποτελείται από 100 πλαίσια σελίδας. Έχουμε επομένως, μια μνήμη των 100000 bytes, που απαρτίζεται από 100 πλαίσια σελίδας των 1000 bytes το κάθε ένα.

Ο πίνακας σελίδων περιέχει μια εγγραφή για κάθε πλαίσιο σελίδας της κύριας μνήμης (δηλαδή είναι ανεστραμμένος) και έχει επομένως 100 εγγραφές. Κάθε εγγραφή του πίνακα σελίδων περιέχει την ακόλουθη πληροφορία:

- τον αριθμό της ιδεατής σελίδας που περιέχει το πλαίσιο,
- έναν μετρητή, ο οποίος αυξάνεται μετά από κάθε πρόσβαση στην αντίστοιχη σελίδα,
- ένα bit τροποποίησης που έχει την τιμή 1 αν η σελίδα τροποποιήθηκε και 0 διαφορετικά.

Ο πίνακας σελίδων μπορεί να οριστεί ως ακολούθως:

```
typedef struct {
```

```
        int page_number;
        int dirty;
        int cnt;
} page_entry_t;
page_entry_t page_table[100];
```

Η ΔΔΣ επικοινωνεί με τις ΔΠΔ μέσω διαμοιραζόμενης μνήμης (κλήσεις συστήματος shmget, shmat, shmctl). Η επίτευξη συγχρονισμού γίνεται με χρήση σημαφόρων (κλήσεις συστήματος semget, semop, semctl). Οι ΔΠΔ μοιράζονται με τη ΔΔΣ μια μεταβλητή access, του ακόλουθου τύπου:

```
typedef struct {
        int address;
        int rw;
} mem_access_t;
mem_access_t access;
```

Η μεταβλητή address στο struct υποδηλώνει την ιδεατή διεύθυνση που θέλει να προσπελάσει η ΔΠΔ, ενώ η μεταβλητή rw παίρνει τιμές 0/1 και υποδηλώνει αν η λειτουργία είναι εγγραφή ή ανάγνωση. Αφού παράγει μια αίτηση προς τη ΔΔΣ, κάθε ΔΠΔ θα πρέπει να αναστέλλεται περιμένοντας τη ΔΔΣ να χειριστεί την αίτηση.

Η ΔΔΣ δε χρειάζεται να προσομοιώνει το γράψιμο και τη μεταφορά σελίδων από και προς το δίσκο. Στην άσκηση αυτή τα περιεχόμενα των σελίδων δε μας ενδιαφέρουν.

Όταν εκκινείται η ΔΔΣ θα πρέπει να αρχικοποιεί τον πίνακα σελίδων, θέτοντας την μεταβλητή page_number κάθε εγγραφής στην τιμή –1, δηλώνοντας έτσι πως το αντίστοιχο πλαίσιο σελίδας είναι κενό. Οι μεταβλητές dirty και cnt κάθε εγγραφής αρχικοποιούνται όταν μια σελίδα τοποθετείται στο πλαίσιο. Τότε, η μεταβλητή cnt παίρνει την τιμή 1, ενώ η rw παίρνει τιμή 1 αν η προσπέλαση στη μνήμη είναι εγγραφή και 0 αν είναι ανάγνωση. Αν το bit τροποποίησης γίνει 1 δεν θα ξαναγίνει 0, παρά μόνο αν η σελίδα αυτή φύγει από τη μνήμη.

Αφού εκτελέσει τις κατάλληλες αρχικοποιήσεις, η ΔΔΣ θα πρέπει να αναστέλλεται έως ότου μια αίτηση γίνει διαθέσιμη (περιγραφή της αίτησης περιέχεται τότε στη μεταβλητή access). Μόλις αυτό συμβεί, ελέγχει αν η σελίδα που σχετίζεται με την ιδεατή διεύθυνση της αίτησης, βρίσκεται στη μνήμη. Αν αυτό συμβαίνει, αυξάνεται απλώς η μεταβλητή cnt της κατάλληλης εγγραφής του πίνακα σελίδων (επίσης, το bit τροποποίησης ενδεχομένως χρειαστεί να αλλάξει από 0 σε 1, αν αυτή η αναφορά είναι η πρώτη εγγραφή στη σελίδα). Διαφορετικά, αναζητείται ελεύθερο πλαίσιο για να τοποθετηθεί η σελίδα. Αν υπάρχει ελεύθερο πλαίσιο η σελίδα τοποθετείται εκεί. Διαφορετικά, κάποια σελίδα θα πρέπει να απομακρυνθεί από τη μνήμη, προκειμένου να τοποθετηθεί στη θέση της η νέα σελίδα. Η σελίδα που επιλέγεται προς απομάκρυνση είναι η σελίδα με το μικρότερο μετρητή, όπως ορίζεται από τον αλγόριθμο αντικατάστασης σελίδων LRU (υποθέτουμε γενική πολιτική τοποθέτησης).

Ο ιδεατός χώρος διευθύνσεων αποτελείται από 500000 bytes και άρα είναι 5 φορές μεγαλύτερος από το μέγεθος της κύριας μνήμης.

Η ΔΔΣ θα πρέπει να διατηρεί τους ακόλουθους μετρητές:

- n1: συνολικός αριθμός αναφορών στη μνήμη,
- n2: συνολικός αριθμός αναφορών που προκάλεσαν απομάκρυνση μιας σελίδας που έχει bit τροποποίησης ίσο με 0,
- n3: συνολικός αριθμός αναφορών που προκάλεσαν απομάκρυνση μιας σελίδας που έχει bit τροποποίησης ίσο με 1.

Κάθε φορά που η συνθήκη [(n1 MODULO 50000) == 0] είναι αληθής, η ΔΔΣ θα πρέπει να εμφανίζει τα ακόλουθα στατιστικά:

1. n2/n1: ρυθμός λαθών σελίδων που δεν χρειάστηκε να γραφούν στο δίσκο,
2. n3/n1:ρυθμός λαθών σελίδων που χρειάστηκε να γραφούν στο δίσκο,
3. (n2+n3)/n1: ρυθμός λαθών σελίδας.

Περιγραφή Διεργασιών Παραγωγής Διευθύνσεων (ΔΠΔ)

Το πρόγραμμα σας θα πρέπει να δημιουργεί 50 ΔΠΔ, που η κάθε μία θα παράγει ιδεατές διευθύνσεις και θα τις κοινοποιεί στην ΔΔΣ. Υποθέστε πως το μέγεθος κάθε διεργασίας είναι 10 σελίδες και πως η $1^{η}$ διεργασία παράγει διευθύνσεις από 0 έως 9999, η $2^{η}$ από 10000 έως 19999, κ.ο.κ. Μια τέτοια διεύθυνση μπορεί να παραχθεί με τον ακόλουθο τρόπο. Η αντίστοιχη διεργασία παράγει έναν τυχαίο αριθμό από 0 έως 9999 και στη συνέχεια του προσθέτει την διεύθυνση βάσης που την χαρακτηρίζει, η οποία είναι 0 για την $1^{η}$ διεργασία, 10000 για τη $2^{η}$, κ.ο.κ.

Για κάθε μία διεύθυνση που θέλει να προσβεί, η ΔΠΔ πρέπει να παράγει έναν ακόμη τυχαίο αριθμό μεταξύ 0 και 1. Αν ο αριθμός αυτός είναι μικρότερος του 0.20, τότε η πρόσβαση στη μνήμη είναι για εγγραφή, δηλαδή 20% των προσβάσεων είναι αιτήσεις εγγραφής. Διαφορετικά, η πρόσβαση στη μνήμη είναι για ανάγνωση.

Κάθε ΔΠΔ τερματίζει αφού αποστείλει στη ΔΔΣ 1000 αιτήσεις.

Παρατηρήσεις

1. *Μπορείτε να δουλέψετε σε ομάδες των δύο ατόμων. Κάθε φοιτητής θα εξεταστεί ωστόσο ατομικά.*
2. Χρησιμοποιείστε την εντολή ipcrm για να σβήνεται τις κοινές μεταβλητές και τους σημαφόρους που δημιουργεί το πρόγραμμά σας. Διαφορετικά σύντομα δεν θα μπορεί να αποδοθεί περισσότερη κοινή μνήμη στο πρόγραμμά σας με αποτέλεσμα αυτό να μην μπορεί να εκτελεστεί.

```
ERROR: syntaxerror
OFFENDING COMMAND: --%ztokenexec_continue--

STACK:

false
```