

**Θέμα 1 (15)**

Σε ένα ηλεκτρονικό σύστημα μεταφοράς κεφαλαίων, κάθε διεργασία κλειδώνει τον λογαριασμό που θα χρεωθεί και τον λογαριασμό που θα πιστωθεί, μεταφέρει τα χρήματα, και αποδεσμεύει τους λογαριασμούς. Υποθέστε ότι εκτελούνται παράλληλα πολλές τέτοιες διεργασίες.

α. Εξηγήστε πως μπορεί να προκληθούν αδιέξοδα.

β. Περιγράψτε έναν τρόπο για την αποφυγή των αδιεξόδων, χωρίς ένας λογαριασμός να αποδεσμεύεται πριν ολοκληρωθεί η δοσοληψία (δεν επιτρέπονται λύσεις που κλειδώνουν ένα λογαριασμό και τον αποδεσμεύουν, αν ο άλλος βρεθεί κλειδωμένος).

**Θέμα 2 (10)**

Έστω ένα πρόγραμμα που αποτελείται από εντολές κάθε μια από τις οποίες πραγματοποιεί 2 προσπελάσεις στη μνήμη. Το 90% των προσπελάσεων γίνονται μέσω συσχετιστικής μνήμης, στην οποία περίπτωση υποθέτουμε ότι δεν υπάρχουν καθυστερήσεις. Όμως, αν μια προσπέλαση γίνει τελικά μέσω του πίνακα σελίδων, τότε υπάρχει καθυστέρηση 0,5μsec. Επιπλέον, η πρόσβαση σε μια σελίδα που δεν βρίσκεται στην μνήμη συνεπάγεται πρόσθετη καθυστέρηση 20msec. Προσδιορίστε το ρυθμό λαθών σελίδας έτσι ώστε ο μέσος χρόνος καθυστέρησης της εκτέλεσης των εντολών λόγω πρόσβασης στην μνήμη να είναι μικρότερος από 0,2μsec.

**Θέμα 3 (15)**

Θεωρείστε ένα σύστημα αρχείων σε ένα δίσκο που έχει τόσο λογικό, όσο και φυσικό μέγεθος μπλοκ 512 bytes. Υποθέστε ότι η πληροφορία διαχείρισης κάθε αρχείου βρίσκεται ήδη στη μνήμη. Για κάθε μια από τις τρεις στρατηγικές ανάθεσης χώρου στο δίσκο (συνεχόμενη, διασυνδεδεμένη, με ευρετήριο) απαντήστε τις ακόλουθες ερωτήσεις:

α. Πως επιτυγχάνεται η απεικόνιση από λογική σε φυσική διεύθυνση; (υποθέστε ότι ο δίσκος αποτελείται από 256 μπλοκ).

β. Αν βρισκόμαστε στο λογικό μπλοκ 10 (το τελευταίο μπλοκ που προσπελάστηκε είναι το 10) και επιθυμούμε να προσπελάσουμε το λογικό μπλοκ 4, πόσα φυσικά μπλοκ πρέπει να διαβαστούν από το δίσκο;

**Θέμα 4 (15)**

Θεωρείστε ένα σύστημα με τρία φυσικά πλαίσια μνήμης στο οποίο δίνεται η παρακάτω ακολουθία αναφορών σελίδων μνήμης:

1, 3, 6, 7, 1, 3, 6, 7, 1, 3, 6, 7, 1, 3, 6, 7.

Ποιο είναι το πλήθος των σφαλμάτων σελίδας που θα συμβούν για κάθε έναν από τους ακόλουθους αλγόριθμους αντικατάστασης σελίδας: (1) το βέλτιστο αλγόριθμο αντικατάστασης σελίδων, (2) τον LRU, (3)

τον αλγόριθμο αντικατάστασης ρολογιού δεύτερης ευκαιρίας. Υπάρχει βέλτιστος αλγόριθμος αντικατάστασης σελίδων που δεν απαιτεί μελλοντική γνώση για κυκλικές ακολουθίες αναφορών όπως αυτή που φαίνεται παραπάνω; Αν ναι δώστε έναν γενικό αλγόριθμο. Αν όχι, εξηγήστε γιατί.

**Θέμα 5 (10)**

Θεωρείστε ένα σύστημα με 64MB φυσικής μνήμης, 32bit φυσικές διευθύνσεις, 32bit εικονικές διευθύνσεις και 4KB πλαίσια φυσικής μνήμης.

α. Χρησιμοποιώντας σχήμα σελιδοποίησης ενός επιπέδου, ποιος είναι ο μέγιστος αριθμός εγγραφών του πίνακα σελίδων για αυτό το σύστημα;

β. Χρησιμοποιώντας σχήμα σελιδοποίησης δύο επιπέδων με εξωτερικό πίνακα σελίδων 1024 εγγραφών, ποια θα ήταν η μετατόπιση σελίδας (page offset) της σελίδας του πίνακα σελίδων που προσπελαύνεται για την εικονική διεύθυνση 0011000000100001110001000011100;

**Θέμα 6 (10)**

Όταν ένας i-node διαβάζεται από το δίσκο κατά τη διάρκεια του ανοίγματος ενός αρχείου, τοποθετείται σε έναν πίνακα i-node στη μνήμη. Αυτός ο πίνακας έχει κάποια πεδία τα οποία δε βρίσκονται στο δίσκο. Ένα από αυτά είναι ένας μετρητής που κρατάει το πλήθος των φορών που έχει ανοιχτεί ο i-node. Εξηγήστε γιατί χρειάζεται ένα τέτοιο πεδίο;

**Θέμα 7 (15)**

Υποθέστε ότι υπάρχουν τέσσερις διεργασίες P1, P2, P3, P4, με χρόνους άφιξης 0, 10, 20, 40, προτεραιότητες 1, 2, 3, 4 και χρόνους εκτέλεσης 30, 20, 50, 20. Οι διεργασίες αυτές χρονοπρογραμματίζονται μέσω μιας ουράς έτοιμων διεργασιών, σε μια μηχανή με έναν επεξεργαστή.

α. Ποιος είναι ο μέσος χρόνος ολοκλήρωσης της εκτέλεσης των παραπάνω διεργασιών με τους εξής αλγορίθμους χρονοπρογραμματισμού: (1) με προτεραιότητες (η υψηλότερη είναι αυτή με τη μικρότερη τιμή), (2) round-robin (ανά 20 μονάδες χρόνου), (3) η μικρότερη διεργασία πρώτα (SJF); Υποθέστε ότι το κόστος εναλλαγής περιβάλλοντος λειτουργίας είναι 1 μονάδα χρόνου, και ότι η άφιξη μιας διεργασίας οδηγεί σε διακοπή που ενεργοποιεί (εκ νέου) τον χρονοπρογραμματισμό.

β. Δίνεται ο παρακάτω ψευδοκώδικας χρονοπρογραμματισμού διεργασιών με προτεραιότητες και διακοπές:

```

create_process(p) {
    ...
    p->counter = p->priority;
    add_to_front_runqueue(p);
    ...
}

schedule() {
    ...
    c = 0;
    for (p = front_runqueue; p <= end_runqueue; p++) {
        if (p->counter > c) {c = p->counter; next = p;}
    }
    if (!c) {
        for (all processes p) {
            p->counter = p->priority;
        }
    }
    run(next);
    ...
}

do_timer_interrupt() {
    ...
    next->counter--;
    if (next->counter < 0) schedule();
    ...
}

```

Υποθέστε ότι κατά την διάρκεια της εκτέλεσης, η `do_timer_interrupt` καλείται σε κάθε διακοπή του χρονομετρητή, ανά 10 μονάδες χρόνου, για να ελέγξει το κατά πόσο πρέπει να επιλεγεί μια άλλη διεργασία. (1) Σχεδιάστε το διάγραμμα Gantt που προκύπτει αν οι διεργασίες χρονοπρογραμματιστούν με αυτή τη μέθοδο, και με βάση τις προτεραιότητες που δόθηκαν παραπάνω. (2)

Ποιος ο μέσος χρόνος αναμονής για τις διεργασίες; Υποθέστε μηδενικό κόστος εναλλαγής περιβάλλοντος λειτουργίας. Υποθέστε επίσης ότι δεν υπάρχουν συνθήκες ανταγωνισμού στον κρίσιμο κώδικα της `do_timer_interrupt`.

**Θέμα 8 (10)**

Ο ελεύθερος χώρος στο δίσκο μπορεί να τηρείται με τη χρήση μιας λίστας ελεύθερων μπλοκ ή ενός bitmap. Οι διευθύνσεις του δίσκου απαιτούν  $D$  bits. Για ένα δίσκο με  $B$  μπλοκ, από τα οποία  $F$  είναι ελεύθερα, διατυπώστε μια συνθήκη που, αν ισχύει, η λίστα ελεύθερων μπλοκ χρησιμοποιεί λιγότερο χώρο από το bitmap. Αν το  $D$  έχει τιμή 16 bits, εκφράστε την απάντησή σας ως ποσοστό του χώρου του δίσκου που πρέπει να είναι ελεύθερος.