

Θέμα 1

Για τον προγραμματισμό αμοιβαίου αποκλεισμού ένα λειτουργικό σύστημα διαθέτει την ατομική εντολή Test-and-Set και σημαφόρους (semaphores). Ποιό το συγκριτικό πλεονέκτημα/μειονέκτημα των δύο μηχανισμών, και πότε είναι προτιμότερη η χρήση του καθενός;

Θέμα 2

Πέντε διεργασίες A, B, Γ, Δ, και E, καταφθάνουν την ίδια χρονική στιγμή (με αυτή τη σειρά) σε ένα υπολογιστικό σύστημα. Οι χρόνοι εκτέλεσης εκτιμώνται σε 10, 6, 2, 4 και 8 λεπτά αντίστοιχα, ενώ οι προτεραιότητες που έχουν δοθεί (στατικά) είναι 3, 5, 2, 1 και 4 αντίστοιχα (το 5 είναι η μεγαλύτερη προτεραιότητα). Σχεδιάστε τα χρονοδιαγράμματα Gantt που δείχνουν την εκτέλεση των διεργασιών για κάθε έναν από τους ακόλουθους αλγόριθμους: (α) εξυπηρέτηση εκ περιτροπής (round robin) με κβάντο χρόνου 1, (β) εξυπηρέτηση με προτεραιότητες, (γ) εξυπηρέτηση κατά σειρά άφιξης (first come first served), και (δ) προτεραιότητα στην συντομότερη διεργασία (shortest job first). Για κάθε αλγόριθμο, υπολογίστε το μέσο χρόνο διεκπεραίωσης του συστήματος καθώς και τον βαθμό καθυστέρησης (% σε σχέση με τον καθαρό χρόνο εκτέλεσης) για κάθε διεργασία. Αγνοείστε την επιβάρυνση λόγω εναλλαγής διεργασιών και υποθέστε πως οι διεργασίες δεν εκτελούν πράξεις εισόδου/εξόδου (I/O).

Θέμα 3

Ένα κλασικό πρόβλημα συγχρονισμού είναι αυτό του ταχυδρομείου, όπου οι πελάτες εξυπηρετούνται από πολλούς ταμίες. Κάθε εισερχόμενος πελάτης παίρνει έναν (αύξοντα) αριθμό και περιμένει μέχρι να κληθεί ο αριθμός του. Κάθε φορά που ένας ταμίας ελευθερώνεται, καλεί και εξυπηρετεί τον πελάτη με τον μικρότερο αριθμό. Γράψτε τον κώδικα για τις διεργασίες πελάτη και ταμιά χρησιμοποιώντας γενικές σημαφόρους (semaphores) έτσι ώστε να γίνεται σωστός συγχρονισμός των αντίστοιχων διεργασιών και οι διεργασίες να ενεργοποιούνται / απενεργοποιούνται κατάλληλα.

Θέμα 4

Ένας υπολογιστής διαθέτει 6 μονάδες CD και N διεργασίες που διεκδικούν ταυτόχρονα την χρήση τους. Κάθε διεργασία χρειάζεται πρόσβαση το πολύ σε 2 CD ταυτόχρονα. Ποιά η μέγιστη τιμή του N για την οποία το σύστημα δεν χρειάζεται να χρησιμοποιήσει αλγόριθμο αποφυγής ή/και ελέγχου αδιεξόδων; Τεκμηριώστε την απάντησή σας. Γιατί τα συνηθισμένα λειτουργικά συστήματα «αγνοούν» το πρόβλημα των αδιεξόδων και δεν χρησιμοποιούν κάποιο μηχανισμό αποφυγής/ελέγχου;

Θέμα 5

Θεωρήστε ένα σύστημα όπου για να εκτελεσθεί μια διεργασία πρέπει πρώτα να μεταφερθεί ολόκληρη στην κυρίως μνήμη. Έστω πως υπάρχουν τα ακόλουθα τμήματα ελεύθερης μνήμης (σε αύξουσα σειρά διευθύνσεων της φυσικής μνήμης): 10K, 4K, 20K, 18K, 7K, 9K, 12K και 15K. Ποιά τμήματα χρησιμοποιούνται και πως τμηματοποιείται περαιτέρω η μνήμη αν φορτωθούν, η μια μετά την άλλη, τρεις διεργασίες που χρειάζονται μνήμη 12K, 10K και 8K αντίστοιχα, με τους αλγόριθμους (α) πρώτης τοποθέτησης (first fit), (β) καλύτερης τοποθέτησης (best fit), και (γ) επόμενης τοποθέτησης (next fit).

Θέμα 6

Έστω ένα σύστημα με εικονικές διευθύνσεις 16 bit, φυσική μνήμη 32K, και σελιδοποίησης ενός επιπέδου. Δίνεται ο πίνακας σελίδων μιας διεργασίας P:

Πίνακας σελίδων	
0	4
1	7
2	άκυρο
3	0
4	άκυρο
5	άκυρο
...	άκυρο
15	άκυρο

Πλαίσια	
0K-4K	→
4K-8K	→
8K-12K	→
12K-16K	→
16K-20K	→
20K-24K	→
24K-28K	→
28K-32K	→

Δώστε τις φυσικές διευθύνσεις που αντιστοιχούν στις εξής εικονικές διευθύνσεις μνήμης της διεργασίας P: 0001110011001101, 0000110011000000, 0011010101010101, 0011110111111111, 0001010101010101 και 0000110011111100. Αν κάθε πρόσβαση στον πίνακα σελίδων απαιτεί 500 nsec και το σύστημα κρατά την τελευταία αναφορά που έγινε στον πίνακα σελίδων σε συσχετική μνήμη 1 θέσης που μπορεί να προσπελασθεί

σε 100 nsec, πόσος χρόνος απαιτείται για την διαδοχική απεικόνιση των παραπάνω εικονικών διευθύνσεων σε φυσικές διευθύνσεις; Υποθέστε πως αρχικά η συσχετική μνήμη είναι άδεια.

Θέμα 7

Ένας υπολογιστής με διευθύνσεις 32 bit χρησιμοποιεί έναν πίνακα σελίδων 2 επιπέδων. Οι εικονικές διευθύνσεις χωρίζονται σε ένα πεδίο 9 bits για τον πίνακα σελίδων πρώτου επιπέδου, σε ένα πεδίο 11 bits για τον πίνακα σελίδων δευτέρου επιπέδου, ενώ τα υπόλοιπα bits υποδεικνύουν την μετατόπιση μέσα στην σελίδα. Πόσο μεγάλες είναι οι σελίδες και πόσες υπάρχουν σε κάθε εικονικό χώρο διευθύνσεων; Γιατί και σε ποιές περιπτώσεις χρησιμοποιούνται πίνακες σελίδων πολλαπλών επιπέδων;

Θέμα 8

Ένας υπολογιστής διαθέτει 4 πλαίσια φυσικής μνήμης. Πόσα λάθη σελίδας θα προκύψουν με την σειρά αναφορών στις σελίδες 0, 1, 7, 2, 3, 2, 7, 1, 0 και 3 για τους αλγόριθμους αντικατάστασης σελίδων (α) FIFO, (β) LRU και (γ) βέλτιστη τοποθέτηση; Υποθέστε πως στην αρχή όλα τα πλαίσια είναι ελεύθερα.

Θέμα 9

Στον οδηγό δίσκου ενός συστήματος H/Y φτάνουν διαδοχικές αιτήσεις για τους κυλίνδρους 10, 22, 20, 2, 40, 6 και 38. Για κάθε μετακίνηση ανάμεσα σε γειτονικούς κυλίνδρους απαιτούνται 6 msec. Ποιός είναι ο συνολικός χρόνος αναζήτησης για τους αλγόριθμους (α) πρώτη εισερχόμενη πρώτη εξυπηρετούμενη (FCFS), (β) προτεραιότητα στον πλησιέστερο κύλινδρο (SSTF), (γ) αλγόριθμος ανελκυστήρα (elevator). Για κάθε περίπτωση, υποθέστε πως αρχικά η κεφαλή είναι τοποθετημένη πάνω από τον κύλινδρο 20.

Θέμα 10

Ένας φίλος σας ισχυρίζεται πως έχει αναπτύξει έναν οδηγό δίσκου για το λειτουργικό Linux, που είναι πολύ γρήγορος γιατί χρησιμοποιεί τον αλγόριθμο του ανελκυστήρα σε συνδυασμό με ουρές αιτήσεων ανά κύλινδρο και σε αύξουσα αριθμητική σειρά τομέα (sector). Εντυπωσιασμένοι κατεβάσετε αμέσως τον κώδικα μέσω διαδικτύου και γράψατε ένα πρόγραμμα που διαβάζει 10000 τυχαία μπλοκ από τον δίσκο. Τρέχετε το πρόγραμμα και διαπιστώνετε πως η επίδοση του προγράμματος είναι ανάλογη με αυτή που θα περίμενε κανείς και από τον αλγόριθμο FCFS (first come first serve). Τελικά μήπως ο φίλος σας έκανε πλάκα; Εξηγήστε γιατί (ναί/όχι).

Θέμα 11

Η απόδοση ενός οδηγού δίσκου εξαρτάται από το ποσοστό επιτυχίας (hit rate) της κρυφής μνήμης (cache) που χρησιμοποιείται για την ενδιάμεση αποθήκευση μπλοκ στην κυρίως μνήμη. Έστω πως για να ικανοποιηθεί μια αίτηση πρόσβασης μέσω κρυφής μνήμης χρειάζεται 1 msec, και για να ικανοποιηθεί μια αίτηση πρόσβασης μέσω δίσκου χρειάζονται 40 msec. Δώστε την μαθηματική συνάρτηση για το μέσο χρόνο που απαιτείται για να ικανοποιηθεί μια αίτηση αν το ποσοστό επιτυχίας είναι h . Σχεδιάστε την καμπύλη της συνάρτησης με βάση τις τιμές της για $h = 0, 1/3, 1/2, 2/3$, και 1. Εξηγήστε γιατί μια σχετικά μικρή σε μέγεθος κρυφή μνήμη συνήθως αρκεί για να επιτευχθεί σχετικά μεγάλη βετίωση στην απόδοση.

Θέμα 12

Δύο προπτυχιακοί φοιτητές H/Y, A και B, συζητούν για το σύστημα αρχείων με ευρετηριακή αποθήκευση που πρέπει να υλοποιήσουν στα πλαίσια της επόμενης εργασίας τους. Ο A υποστηρίζει πως καθώς η κυρίως μνήμη είναι φτηνή και αρκετή, όταν το σύστημα λαμβάνει αίτηση να ανοίξει ένα αρχείο, δεν χρειάζεται να διανύει την λίστα των αρχείων του συστήματος για να βρεί (αν υπάρχει) τον κόμβο που περιέχει το αντίστοιχο ευρετήριο, αλλά είναι πιο απλό και γρήγορο να δημιουργεί ένα νέο αντίγραφο του ευρετηρίου του αρχείου και να τον τοποθετεί στην λίστα των ανοιχτών αρχείων. Ο B διαφωνεί. Ποιός έχει δίκιο και γιατί;