



Ενότητα 4: Η Ιδεατή μνήμη και η Προστασία της

Σκοπός Ο σκοπός της ενότητας αυτής είναι να παρουσιάσει την ιδεατή μνήμη, τον τρόπο λειτουργίας της, καθώς και τον τρόπο με τον οποίο αυτή χρησιμοποιείται.

Προσδοκώμενα Αποτελέσματα Όταν θα έχετε μελετήσει την ενότητα, θα είστε σε θέση να:



περιγράφετε την τεχνική της ιδεατής μνήμης,



συγκρίνετε την ιδεατή με την κρυφή μνήμη,



κατηγοριοποιείτε τα συστήματα της ιδεατής μνήμης,



προσδιορίζετε τις τεχνικές για γρήγορη μετάφραση διευθύνσεων,



επιλέγετε το κατάλληλο μέγεθος σελίδας της ιδεατής μνήμης,



περιγράφετε τον τρόπο με τον οποίο επιτυγχάνεται η προστασία σε περιβάλλοντα πολυπρογραμματισμού,



απαριθμείτε τις προϋποθέσεις για την προστασία των διεργασιών.



ιδεατή μνήμη, ιδεατές διευθύνσεις, σελίδα, τμήμα, σελιδοποίηση, τμηματοποίηση, πίνακας σελίδων, καταχωρητής μετάφρασης, bit, αναφοράς ή bit χρήσης, δυαδικό ψηφίο τροποποίησης, εσωτερικός κατακερματισμός, πολυπρογραμματισμός, διεργασία, πολλαπλή προσπέλαση, σημαίες επίτρεψης, κλειδιά προστασίας



Η Ιδεατή μνήμη

Για να εκτελεστεί ένα πρόγραμμα είναι απαραίτητο να φορτωθεί στην κύρια μνήμη του υπολογιστή. Όταν το πρόγραμμα είναι μικρότερο από το μέγεθος της κύριας μνήμης, τότε δεν υπάρχει κανένα πρόβλημα. Πρόβλημα υπάρχει, όταν το πρόγραμμα είναι μεγαλύτερο από την κύρια μνήμη ή από το τμήμα της κύριας μνήμης που διατίθεται γι' αυτό (στην περίπτωση του πολυπρογραμματισμού).



Στα σύγχρονα συστήματα υπολογιστών χρησιμοποιείται η τεχνική της ιδεατής μνήμης, η οποία υλοποιείται με τη χρήση ενός τμήματος του σκληρού δίσκου σαν μία λογική επέκταση της κύριας μνήμης.

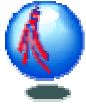
Το Λειτουργικό Σύστημα σε συνεργασία με το κατάλληλο υλικό κρατάει στην κύρια μνήμη εκείνα τα τμήματα του προγράμματος που χρησιμοποιούνται και τα υπόλοιπα τα διατηρεί στο σκληρό δίσκο. Τα τμήματα αυτά μεταφέρονται στην κύρια μνήμη, όταν ζητηθούν. Αντίστοιχα τμήματα της κύριας μνήμης μπορούν να μεταφερθούν στο δίσκο, όταν δε χρησιμοποιούνται. Για την επιλογή των τμημάτων που θα μεταφερθούν χρησιμοποιούνται ειδικοί αλγόριθμοι. Η διαδικασία αυτή ονομάζεται **εναλλαγή**.

Η τεχνική αυτή απαλλάσσει τον προγραμματιστή από την ανάγκη να χωρίσει το πρόγραμμα του σε τμήματα ικανά να μεταφερθούν σε συγκεκριμένο μέγεθος της κύριας μνήμης. Τα προγράμματά χρησιμοποιούν ιδεατές διευθύνσεις, δηλαδή απευθύνονται σε μία ιδεατή μνήμη η οποία έχει πολύ μεγαλύτερο μέγεθος από την κύρια μνήμη ενός υπολογιστή. Όταν το πρόγραμμα εκτελεστεί, τότε το Λειτουργικό Σύστημα με τη συνεργασία του υλικού αναλαμβάνει να μετατρέψει τις ιδεατές διευθύνσεις σε πραγματικές.



ΔΡΑΣΤΗΡΙΟΤΗΤΑ 1

Να εξηγήσετε σε 10 περίπου γραμμές τι είναι η ιδεατή μνήμη και για ποιο λόγο χρησιμοποιείται στους υπολογιστές. Να συγκρίνετε την απάντησή σας με την υποενότητα: «Η ιδεατή μνήμη».



Σύγκριση ιδεατής μνήμης και κρυφής μνήμης

Οι βασικές διαφορές ανάμεσα στην κρυφή και την ιδεατή μνήμη είναι:

- Η διαδικασία της αντικατάστασης στις αποτυχίες της κρυφής μνήμης ελέγχεται κυρίως από το υλικό ενώ στην ιδεατή από το λειτουργικό σύστημα. Επειδή η ποινή αποτυχίας της ιδεατής μνήμης είναι κατά πολύ μεγαλύτερη από αυτή της κρυφής μνήμης, είναι σκόπιμο να αφιερωθεί από το λειτουργικό σύστημα περισσότερος χρόνος, ώστε να πάρει μια καλή απόφαση για την αντικατάσταση.
- Το μέγεθος των διευθύνσεων που χρησιμοποιεί ο επεξεργαστής καθορίζει το μέγεθος της ιδεατής μνήμης ενώ το μέγεθος της κρυφής μνήμης είναι ανεξάρτητο από το μήκος των διευθύνσεων μνήμης.
- Η δευτερεύουσα αποθήκευση αντιστοιχεί στο χαμηλότερο επίπεδο μνήμης στην ιεραρχία. Επιπρόσθετα, χρησιμοποιείται και από το σύστημα αρχείων, που καταλαμβάνει το μεγαλύτερο μέρος της και συνήθως δεν είναι τμήμα του χώρου διευθύνσεων.

► Συστήματα ιδεατής μνήμης

Τα συστήματα ιδεατής μνήμης υποδιαιρούνται σε δυο κλάσεις: Σε εκείνα που έχουν καθορισμένο μέγεθος μπλοκ που ονομάζεται **σελίδα** και σε εκείνα που έχουν μεταβλητό μέγεθος μπλοκ που ονομάζονται **τμήματα**. Το σχήμα 4.4.1 δείχνει ένα παράδειγμα των δύο παραπάνω προσεγγίσεων.



Σχήμα 4.4.1 - Ο κώδικας και τα δεδομένα στη σελιδοποίηση χρησιμοποιούν σταθερού μεγέθους μπλοκ, σε αντίθεση με τη τμηματοποίηση, όπου το μέγεθος ανά μπλοκ μπορεί να διαφέρει.

Η σελιδοποιημένη διευθυνσιοδότηση χρησιμοποιεί μία απλή σταθερού μεγέθους διεύθυνση, διαιρεμένη σε αριθμό σελίδας και σε θέση μέσα στη σελίδα. Η κατατμημένη διευθυνσιοδότηση απαιτεί μια λέξη για τον αριθμό τμήματος και μια λέξη για τη θέση μέσα στο τμήμα, λόγω του μεταβλητού μεγέθους των τμημάτων. Έτσι ένας σελιδοποιημένος χώρος διευθύνσεων είναι απλούστερος για το μεταγλωττιστή. Ορισμένες υλοποιήσεις συστημάτων υπολογιστών χρησιμοποιούν μια μεικτή προσέγγιση η οποία καλείται σελιδοποιημένα τμήματα. Η προσέγγιση αυτή αντιμετωπίζει καλύτερα το πρόβλημα που δημιουργούν οι αντικαταστάσεις των μπλοκ δεδομένων.

❖ Τοποθέτηση ενός μπλοκ στην κύρια μνήμη

Η ποινή αποτυχίας της ιδεατής μνήμης περιλαμβάνει πρόσβαση σε μια περιστρεφόμενη μαγνητική συσκευή αποθήκευσης (συνήθως ένα σκληρό δίσκο) και γι' αυτό είναι αρκετά υψηλή. Δίνοντας τη δυνατότητα επιλογής ανάμεσα σε χαμηλότερους ρυθμούς αποτυχίας και απλούστερους αλγόριθμους αντικατάστασης, οι σχεδιαστές λειτουργικών συστημάτων διαλέγουν πάντα τους χαμηλότερους ρυθμούς αποτυχίας. Επομένως τα λειτουργικά συστήματα σχεδιάζονται με τέτοιο τρόπο, ώστε να επιτρέπεται η τοποθέτηση ενός μπλοκ σε οποιαδήποτε διεύθυνση της κύριας μνήμης.

❖ Εντοπισμός μπλοκ στην κύρια μνήμη

Τόσο η μέθοδος της σελιδοποίησης όσο και της τμηματοποίησης στηρίζονται σε μία δομή δεδομένων που για να προσπελαστεί, χρειάζεται ο αριθμός σελίδας ή ο αριθμός τμήματος. Η δομή αυτή περιέχει την φυσική διεύθυνση του μπλοκ. Στην τμηματοποίηση η μετατόπιση προστίθεται στη φυσική διεύθυνση του τμήματος και παράγεται η τελική ιδεατή διεύθυνση. Στη σελιδοποίηση η διεύθυνση είναι απλά η φυσική διεύθυνση της σελίδας.

Η δομή αυτή που περιέχει τις φυσικές διευθύνσεις των σελίδων συνήθως έχει τη μορφή ενός **πίνακα σελίδων**. Το μέγεθος του πίνακα στον οποίο αναφέρεται κανείς με τον ιδεατό αριθμό σελίδας, είναι ίσο με τον αριθμό των σελίδων στο ιδεατό πεδίο διευθύνσεων. Για να μειωθεί το μέγεθος του πίνακα, μερικοί υπολογιστές εφαρμόζουν μία συνάρτηση σύνοψης στις ιδεατές διευθύνσεις, έτσι ώστε ο πίνακας να έχει μέγεθος ίσο με τον αριθμό των φυσικών διευθύνσεων της κύριας μνήμης, ο οποίος είναι πολύ μικρότερος από τον αριθμό των ιδεατών σελίδων. Η δομή αυτή ονομάζεται **αντιστρεπτός πίνακας σελίδων**. Για να μειωθεί ο χρόνος της μετάφρασης των διευθύνσεων, χρησιμοποιείται μια κρυφή μνήμη αφιερωμένη στη μετάφραση των διευθύνσεων και ονομάζεται **καταχωρητής μετάφρασης**.

❖ Αντικατάσταση μπλοκ σε μια αποτυχία ιδεατής μνήμης

Σχεδόν όλα τα λειτουργικά συστήματα χρησιμοποιούν την τεχνική αντικατάστασης του λιγότερο πρόσφατα χρησιμοποιημένου μπλοκ -LRU τεχνική τοποθέτησης- με τη λογική ότι το μπλοκ αυτό δε θα χρησιμοποιηθεί στο άμεσο μέλλον. Για την υλοποίηση αυτού του αλγορίθμου ορισμένα υπολογιστικά συστήματα παρέχουν ένα **bit χρήσης** ή **bit αναφοράς** για κάθε σελίδα. Το bit αναφοράς τίθεται όταν η σελίδα προσπελαύνεται. Το λειτουργικό σύστημα περιοδικά καταγράφει και καθαρίζει τα bits αυτά, ώστε να μπορεί να ξέρει σε ποιες σελίδες έγινε αναφορά κατά τη διάρκεια μιας συγκεκριμένης χρονικής περιόδου.

❖ Εγγραφές στη μνήμη

Το αμέσως χαμηλότερο επίπεδο από τη μνήμη στην ιεραρχία, περιέχει περιστρεφόμενους μαγνητικούς δίσκους τους οποίους για να προσπελάσεις, χρειάζεσαι εκατομμύρια κύκλους ρολογιού. Εξαιτίας των πολύ μεγάλων διαφορών στους χρόνους προσπέλασης μνήμης και δίσκου, δεν υπάρχει λειτουργικό σύστημα με ιδεατή μνήμη το οποίο να μπορεί να γράφει κατευθείαν στο δίσκο τις αλλαγές που προκαλεί η ΚΜΕ στα δεδομένα της μνήμης. Έτσι για την εγγραφή δεδομένων χρησιμοποιείται πάντα η τεχνική της επανεγγραφής. Λόγω του υψηλού χρονικού κόστους της προσπέλασης στο δίσκο, τα συστήματα ιδεατής μνήμης συνήθως συμπεριλαμβάνουν ένα δυαδικό ψηφίο, το bit τροποποίησης ή dirty bit, δηλωτικό της τροποποίησης ενός μπλοκ. Κατά συνέπεια, τα μόνα μπλοκ που επανεγγράφονται στο δίσκο είναι εκείνα τα οποία έχουν τροποποιηθεί.



ΔΡΑΣΤΗΡΙΟΤΗΤΑ 2

Να αναφέρετε τις βασικές διαφορές ανάμεσα στην κρυφή και την ιδεατή μνήμη. Να συγκρίνετε την απάντησή σας με την πρώτη παράγραφο της υποενότητας που μόλις διαβάσατε.



ΔΡΑΣΤΗΡΙΟΤΗΤΑ 3

Μπορείτε να εξηγήσετε το λόγο για τον οποίο υπάρχουν δύο διαφορετικά συστήματα ιδεατής μνήμης; Να αιτιολογήσετε την απάντησή σας.



ΑΠΑΝΤΗΣΗ ΔΡΑΣΤΗΡΙΟΤΗΤΑΣ 3

Κάθε ένα από τα δύο συστήματα μπορεί να είναι κατάλληλο για συγκεκριμένες εφαρμογές. Για παράδειγμα, στην περίπτωση μιας βάσης δεδομένων όπως ένας τηλεφωνικός κατάλογος, τα δεδομένα και επομένως τα μπλοκ πρέπει να είναι σταθερού μήκους (σελιδοποίηση). Σε κάποια άλλη περίπτωση όπου τα δεδομένα δεν έχουν το ίδιο μέγεθος, με τη σελιδοποίηση θα έπρεπε να ψάξουμε σε δύο ή τρία μπλοκ για να βρούμε τα δεδομένα. Αν το μέγεθος των μπλοκ στη σελιδοποίηση ήταν ίσο με το μεγαλύτερο μπλοκ, τότε θα υπήρχε σπατάλη χώρου. Επομένως η επιλογή ανάμεσα στα δύο συστήματα εξαρτάται από την υλοποίηση της εκάστοτε εφαρμογής.



Τεχνικές για γρήγορη μετάφραση διευθύνσεων

Οι πίνακες σελίδων συνήθως είναι μεγάλοι, αποθηκεύονται στην κύρια μνήμη και μερικές φορές είναι και οι ίδιοι σελιδοποιημένοι. Κατά συνέπεια, κάθε πρόσβαση στη μνήμη χρειάζεται τουλάχιστον διπλάσιο χρόνο, μία πρόσβαση για την εύρεση της φυσικής διεύθυνσης και μια δεύτερη για τα ίδια τα δεδομένα. Ως λογική απόρροια, το κόστος είναι πολύ μεγάλο.

Μία λύση είναι να αποθηκεύεται η τελευταία μετάφραση διεύθυνσης, έτσι ώστε να μην χρειάζεται να γίνει νέα μετάφραση αν η τρέχουσα διεύθυνση αναφέρεται στην ίδια σελίδα με την προηγούμενη διεύθυνση.

Μία άλλη λύση αξιοποιεί την αρχή της τοπικότητας. Αν οι προσβάσεις έχουν τοπικότητα, τότε και οι μεταφράσεις των διευθύνσεων θα έχουν τοπικότητα. Κρατώντας αυτές τις μεταφράσεις διευθύνσεων σε έναν ειδικό καταχωρητή, μια πρόσβαση στη μνήμη σπάνια χρειάζεται μια δεύτερη πρόσβαση για την μετάφραση των δεδομένων. Αυτός ο ειδικός καταχωρητής, ονομάζεται translation look-aside buffer ή TLB.

Μια καταχώρηση στον TLB είναι σαν την καταχώρηση της κρυφής μνήμης όπου η ετικέτα κρατά τμήματα της ιδεατής διεύθυνσης και το τμήμα δεδομένων κρατά έναν αριθμό φυσικού πλαισίου σελίδας, ένα πεδίο προστασίας, ένα δυαδικό ψηφίο εγκυρότητας, ένα δυαδικό ψηφίο ένδειξης χρήσης και ένα δυαδικό ψηφίο τροποποίησης. Για ν' αλλάξει τον αριθμό φυσικού πλαισίου σελίδας ή το bit προστασίας μιας καταχώρησης στον πίνακα σελίδων, το λειτουργικό σύστημα πρέπει να σιγουρευτεί ότι η παλιά καταχώρηση δεν βρίσκεται στον TLB. Διαφορετικά το σύστημα δεν θα συμπεριφερθεί σωστά. Το δυαδικό ψηφίο τροποποίησης δηλώνει ότι η σελίδα που πήγε να ταιριαστεί έχει τροποποιηθεί και όχι ότι η μετάφραση της διεύθυνσης στον TLB ή ότι ένα συγκεκριμένο μπλοκ στην κρυφή μνήμη δεδομένων έχουν τροποποιηθεί.



Επιλογή μεγέθους σελίδας

Η πιο χρήσιμη παράμετρος από πλευράς αρχιτεκτονικής είναι το μέγεθος σελίδας. Η επιλογή της σελίδας είναι θέμα ισορροπίας παραμέτρων, ανάμεσα σ' εκείνες οι οποίες ευνοούν ένα μεγαλύτερο μέγεθος σελίδας και σ' εκείνες που ευνοούν ένα μικρότερο μέγεθος. Οι λόγοι που συνηγορούν υπέρ του μεγαλύτερου μεγέθους είναι:

- Το μέγεθος του πίνακα σελίδων είναι αντιστρόφως ανάλογο του μεγέθους της σελίδας. Μπορεί επομένως να γίνει λιγότερη σπατάλη μνήμης φτιάχνοντας τις σελίδες μεγαλύτερες.
- Ένα μεγαλύτερο μέγεθος απλοποιεί τους χρόνους επιτυχούς λειτουργίας της γρήγορης κρυφής μνήμης.
- Η μεταφορά μεγαλύτερων σελίδων από και προς τη δευτερεύουσα αποθήκευση, πιθανά μέσα από δίκτυο, είναι πιο αποδοτική από τη μεταφορά μικρότερων.
- Ο αριθμός των καταχωρήσεων στον TLB είναι περιορισμένος. Οπότε ένα μεγαλύτερο μέγεθος σελίδας σημαίνει ότι περισσότερη μνήμη μπορεί να απεικονιστεί αποδοτικότερα και έτσι μειώνεται ο αριθμός των αποτυχιών του TLB.

Το κύριο κίνητρο για μικρότερο μέγεθος σελίδας είναι η συντηρητική αποθήκευση. Ένα μικρό μέγεθος σελίδας θα έχει ως αποτέλεσμα λιγότερο χαμένο χώρο αποθήκευσης όταν μια ενιαία περιοχή της ιδεατής μνήμης δεν είναι ίση στο μέγεθος με ένα πολλαπλάσιο του μεγέθους σελίδας. Ο όρος γι' αυτήν την αχρησιμοποίητη μνήμη σε μια σελίδα είναι **εσωτερικός κατακερματισμός**.

Μια τελευταία επίπτωση είναι ο χρόνος εκκίνησης μιας διεργασίας. Πολλές διεργασίες είναι μικρές κι έτσι μεγαλύτερα μεγέθη σελίδας θα επιμήκυναν το χρόνο που απαιτείται για να κληθεί μια διεργασία.



ΔΡΑΣΤΗΡΙΟΤΗΤΑ 4

Να προσδιορίσετε τις διαφορές της σελιδοποίησης και της τμηματοποίησης και να συμπληρώσετε τον παρακάτω πίνακα:

	Σελιδοποίηση	Τμηματοποίηση
Λέξεις ανά διεύθυνση		
Ορατό στον προγραμματιστή		
Χρήση του μεγέθους της μνήμης		
Αποδοτική χρήση δίσκου		



ΑΠΑΝΤΗΣΗ ΔΡΑΣΤΗΡΙΟΤΗΤΑΣ 4

	Σελιδοποίηση	Τμηματοποίηση
Λέξεις ανά διεύθυνση	Μία	Δύο (τμήμα και μετατόπιση)
Ορατό στον προγραμματιστή	Αθέατο στον προγραμματιστή εφαρμογών	Μπορεί να είναι ορατό
Αντικατάσταση μπλοκ	Εύκολη	Δύσκολη (πρέπει να βρεθεί συγκεκριμένου μεγέθους τμήμα στη μνήμη)
Χρήση του μεγέθους της μνήμης	Εσωτερικός κατακερματισμός (υπάρχουν τμήματα σελίδας που δεν χρησιμοποιούνται)	Εξωτερικός κατακερματισμός (υπάρχουν τμήματα στη μνήμη που δεν χρησιμοποιούνται)
Αποδοτική χρήση δίσκου	Ναι (το μέγεθος σελίδας επιλέγεται ανάλογα)	Όχι πάντα (τα μικρά τμήματα μπορεί να μεταφέρουν πολύ λίγα δεδομένα)



Προστασία σε περιβάλλοντα πολυπρογραμματισμού

Η υλοποίηση του **πολυπρογραμματισμού**, όπου ένας υπολογιστής είναι «μοιρασμένος» σε πολλά προγράμματα τα οποία τρέχουν παράλληλα, οδήγησε σε νέες απαιτήσεις όσον αφορά στην προστασία των προγραμμάτων. Κάθε πρόγραμμα μαζί με οποιαδήποτε κατάσταση χρειάζεται αυτό για να συνεχίσει να εκτελείται, ονομάζεται **διεργασία**. Ο πολυπρογραμματισμός παρέχει τη δυνατότητα σε διαφορετικούς χρήστες να έχουν πρόσβαση στον υπολογιστή ταυτόχρονα, δίνοντας τη ψευδαίσθηση ότι όλοι έχουν τον πλήρη έλεγχο του συστήματος. Επομένως σε κάθε στιγμή πρέπει να είναι δυνατή η **εναλλαγή** ανάμεσα στις διεργασίες. Αυτός ο τρόπος λειτουργίας ανάμεσα σε διαλογικούς χρήστες ονομάζεται **πολλαπλή προσπέλαση ή καταμερισμός χρόνου**.

Σε περιβάλλον πολυπρογραμματισμού, πρέπει να διασφαλίσουμε ότι κάθε διεργασία θα ολοκληρωθεί σωστά. Ο σχεδιαστής του υπολογιστή θα διασφαλίσει ότι το κομμάτι της ΚΜΕ που διατηρεί την κατάσταση της διεργασίας, μπορεί να σωθεί και να καταγραφεί. Ο σχεδιαστής του λειτουργικού συστήματος θα εγγυηθεί ότι οι διεργασίες δεν παρεμβαίνουν η μία στους υπολογισμούς της άλλης. Το ζήτημα της προστασίας της κατάστασης μιας διεργασίας αντιμετωπίζεται με την τμηματοποίηση της κύριας μνήμης από το λειτουργικό σύστημα. Κατά συνέπεια, πολλές, διαφορετικές διεργασίες έχουν τη δική τους κατάσταση στη μνήμη την ίδια χρονική στιγμή.



Πέρα από την προστασία, οι υπολογιστές παρέχουν διαμοιρασμό κώδικα και δεδομένων, για να επιτρέψουν την επικοινωνία ανάμεσα στις διεργασίες ή να κάνουν οικονομία μνήμης, με το να μειώσουν τον αριθμό της ίδιας πληροφορίας.



Η προστασία των διεργασιών

Ο απλούστερος μηχανισμός προστασίας χρησιμοποιεί ένα ζευγάρι καταχωρητών που κρατούν δύο πεδία, τη **βάση** και το **όριο**. Κάθε διεύθυνση ελέγχεται και θεωρείται έγκυρη αν: **Βάση \leq διεύθυνση \leq Όριο**

Σε μερικά συστήματα η διεύθυνση θεωρείται ως ένας μη προσημασμένος αριθμός, ο οποίος προστίθεται πάντα στη βάση, οπότε: **(Βάση + διεύθυνση) \leq Όριο**

Οι διεργασίες χρήστη δεν είναι ελεύθερες να αλλάζουν τη βάση και το όριο. Το λειτουργικό σύστημα, ωστόσο, έχει αυτή τη δυνατότητα να μπορεί να εναλλάσσει τις διεργασίες. Επομένως ο σχεδιαστής του υπολογιστή για να προστατέψει τις διεργασίες, πρέπει να παρέχει ορισμένες προϋποθέσεις στο σχεδιαστή του λειτουργικού συστήματος:

1. Να παρέχει δύο καταστάσεις, οι οποίες υποδεικνύουν αν η τρέχουσα διεργασία είναι μια διεργασία χρήστη ή μια διεργασία του λειτουργικού συστήματος (κελύφους ή επόπτη).
2. Να παρέχει ένα τμήμα της κατάστασης της ΚΜΕ, το οποίο μια διεργασία να μπορεί να χρησιμοποιήσει αλλά όχι να γράψει. Αυτό συμπεριλαμβάνει τους καταχωρητές **βάσης-ορίου**, το δυαδικό ψηφίο κατάστασης **χρήστη-επόπτη** και το ψηφίο **ενεργοποίησης-απενεργοποίησης** εξαιρέσεων.
3. Να παρέχουν μηχανισμούς, σύμφωνα με τους οποίους η ΚΜΕ να μπορεί να μεταβαίνει από την κατάσταση χρήστη στην κατάσταση επόπτη και αντίστροφα.

Ενώ η βάση και το όριο συνιστούν το ελάχιστο σύστημα προστασίας, η ιδεατή μνήμη προσφέρει μια καλύτερη λύση. Όπως έχουμε αναφέρει, οι διευθύνσεις περνούν μέσω μιας απεικόνισης από την ιδεατή στη φυσική μορφή τους. Σύμφωνα με αυτή την τεχνική, κάθε διεργασία διαθέτει ένα πίνακα σελίδων, ο οποίος απεικονίζει τις διακριτές σελίδες στη μνήμη που μπορεί να προσπελάσει. Αυτή η προσέγγιση μπορεί να παρέχει προστασία χρήστη-επόπτη και ανάγνωσης-εγγραφής, απλά χρησιμοποιώντας **σημαίες επίτρεψης** σε κάθε σελίδα στη μνήμη. Σε περίπτωση που το υλικό ανιχνεύσει μη εξουσιοδοτημένες προσβάσεις στη μνήμη διακόπτει τη λειτουργία της ΚΜΕ και καλεί το λειτουργικό σύστημα πριν προκαλέσουν δυσλειτουργία. Προφανώς τα προγράμματα χρήστη δεν μπορούν να τροποποιήσουν τους πίνακες σελίδων τους.



Η προστασία χρήστη-επόπτη μπορεί να κλιμακωθεί σε περισσότερα επίπεδα, ανάλογα με την ανησυχία του σχεδιαστή. Παράδειγμα αποτελεί η δομή προστασίας με δακτυλίους, που ακολουθείται από τον Intel Pentium. Στους εσωτερικούς δακτυλίους έχουν πρόσβαση μόνο εξουσιοδοτημένα άτομα, και όσο πιο εσωτερικός είναι ένας δακτύλιος τόσο πιο μεγάλη εξουσιοδότηση απαιτείται.

Μια περαιτέρω αύξηση της προστασίας παρέχουν τα **κλειδιά προστασίας**, όπου ένας χρήστης ή ένα πρόγραμμα δεν μπορεί να έχει πρόσβαση σε δεδομένα που έχουν κλειδωθεί, εκτός αν έχει το κατάλληλο κλειδί. Σε αυτή την περίπτωση το λειτουργικό σύστημα και το

υλικό πρέπει να είναι ικανά να περνούν τα κλειδιά ρητά από το ένα πρόγραμμα στο άλλο, χωρίς να επιτρέπουν στο ίδιο το πρόγραμμα να τα αντιγράψει. Τέτοιος έλεγχος απαιτεί σημαντική υποστήριξη από το υλικό, προκειμένου ο χρόνος για τον έλεγχο των κλειδιών να παραμείνει χαμηλός.



ΔΡΑΣΤΗΡΙΟΤΗΤΑ 5

Να εξηγήσετε με ποιον τρόπο παρέχεται προστασία των διεργασιών με τη χρήση της ιδεατής μνήμης.



ΑΠΑΝΤΗΣΗ ΔΡΑΣΤΗΡΙΟΤΗΤΑΣ 5

Με τη χρήση της ιδεατής μνήμης, οι διευθύνσεις περνούν μέσω μιας απεικόνισης από την ιδεατή στη φυσική μορφή τους. Κάθε διεργασία διαθέτει ένα πίνακα σελίδων, ο οποίος απεικονίζει τις διακριτές σελίδες στη μνήμη που μπορεί να προσπελάσει. Αυτή η προσέγγιση μπορεί να παρέχει προστασία χρήστη/επόπτη και ανάγνωσης/εγγραφής, απλά χρησιμοποιώντας σημαίες επίτρεψης σε κάθε σελίδα στη μνήμη. Σε περίπτωση που το υλικό ανιχνεύσει μη εξουσιοδοτημένες προσβάσεις στη μνήμη διακόπτει τη ΚΜΕ και καλεί το λειτουργικό σύστημα πριν μπορέσουν να κάνουν ζημιά. Προφανώς τα προγράμματα χρήστη δεν μπορούν να τροποποιήσουν τους πίνακες σελίδων τους.



Ανακεφαλαιώνοντας λοιπόν...



Η τεχνική της ιδεατής μνήμης υλοποιείται με τη χρήση ενός τμήματος του σκληρού δίσκου σαν μία λογική επέκταση της κύριας μνήμης. Τα συστήματα ιδεατής μνήμης χρησιμοποιούν είτε σελιδοποιημένη είτε κατατμημένη διευθυνσιοδότηση.



Οι βασικές διαφορές της κρυφής με την ιδεατή μνήμη έχουν σχέση με: τη διαδικασία της αντικατάστασης στις αποτυχίες, το μέγεθος των διευθύνσεων που χρησιμοποιεί ο επεξεργαστής και με το επίπεδο στην ιεραρχία της μνήμης.



Υπάρχουν διάφορες τεχνικές για γρήγορη μετάφραση διευθύνσεων. Μία είναι η αποθήκευση της τελευταίας μετάφρασης διεύθυνσης και μία άλλη αξιοποιεί την αρχή της τοπικότητας, κρατώντας τις μεταφράσεις διευθύνσεων σε ειδικό καταχωρητή (TLB).



Η επιλογή της σελίδας είναι θέμα ισορροπίας παραμέτρων, ανάμεσα σ' εκείνες οι οποίες ευνοούν ένα μεγαλύτερο μέγεθος σελίδας, π.χ απλοποιεί τους χρόνους επιτυχούς λειτουργίας της γρήγορης κρυφής μνήμης, και σ' εκείνες που ευνοούν ένα μικρότερο μέγεθος, π.χ οι μικρές σελίδες περιορίζουν το φαινόμενο του εσωτερικού κατακερματισμού.



Σε περιβάλλον πολυπρογραμματισμού το ζήτημα της προστασίας της κατάστασης μιας διεργασίας αντιμετωπίζεται με την τμηματοποίηση της κύριας μνήμης από το λειτουργικό σύστημα. Κατά συνέπεια, πολλές, διαφορετικές διεργασίες έχουν τη δική τους κατάσταση στη μνήμη την ίδια χρονική στιγμή.



Για να προστατέψω μία διεργασία, μπορώ να τμηματοποιήσω την κύρια μνήμη απ'το λειτουργικό σύστημα. Σαν αποτέλεσμα, διαφορετικές διεργασίες έχουν τη δική τους κατάσταση στη μνήμη την ίδια χρονική στιγμή.