



Λειτουργικά Συστήματα

Ενότητα 5α: Διαχείριση Κύριας Μνήμης

Αθηνά Βακάλη
Τμήμα Πληροφορικής



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο

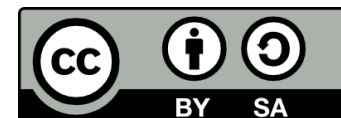


ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΣΠΑ
2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ



Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.

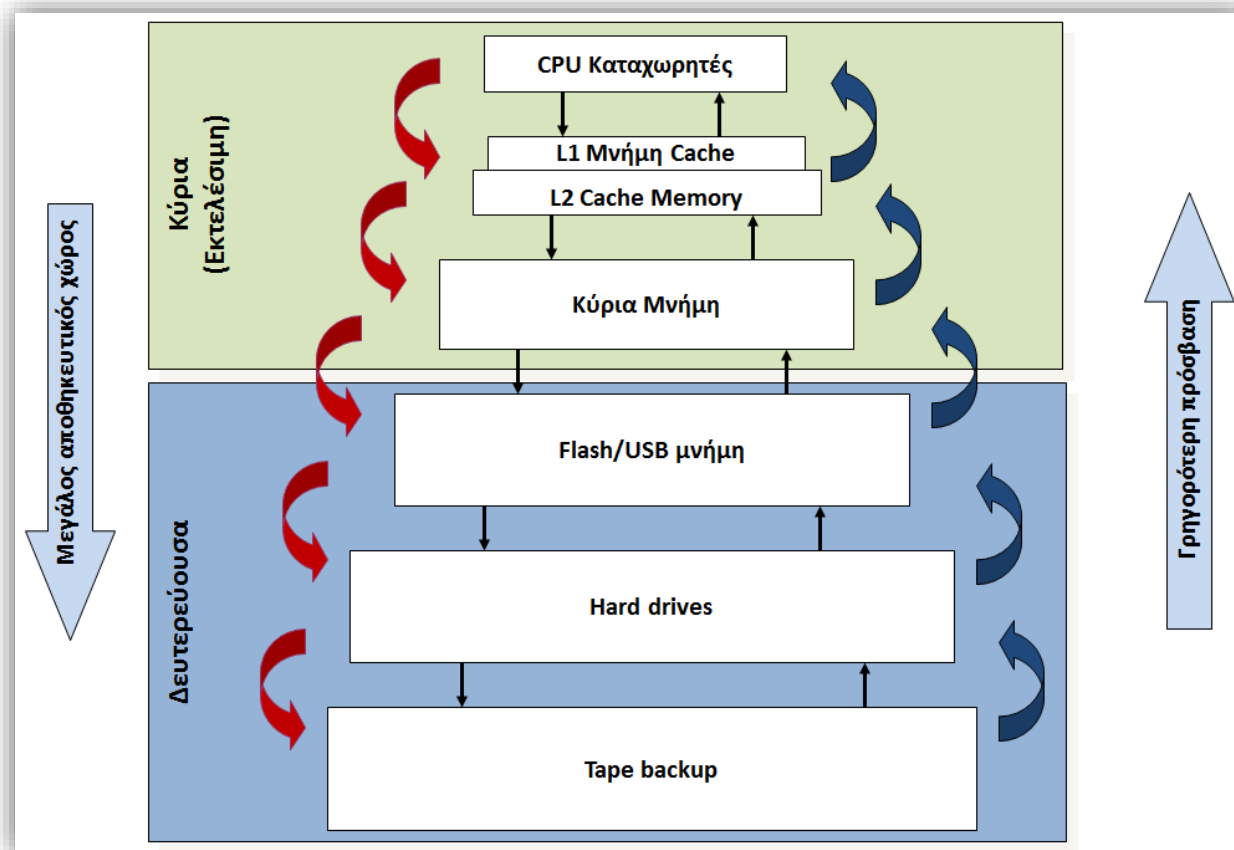


Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «Ανοικτά Ακαδημαϊκά Μαθήματα στο Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.



Ιεραρχία Μνήμης



Ιεραρχία μνήμης

Διαχείριση Μνήμης

Η **διαχείριση μνήμης** είναι η λειτουργία της υποδιαίρεσης της μνήμης από το λειτουργικό σύστημα με δυναμικό τρόπο ώστε να εξυπηρετούνται όσο το δυνατόν περισσότερες διεργασίες.



Διαχειριστής Μνήμης

Το τμήμα του Λειτουργικού Συστήματος που ασχολείται με την οργάνωση και τις στρατηγικές διαχείρισης της ιεραρχίας μνήμης.

Αποστολή

- Να παρακολουθεί ποια τμήματα της μνήμης χρησιμοποιούνται και ποια όχι.
- Να κατανέμει τη μνήμη στις διεργασίες και να την παίρνει πίσω όταν δεν τη χρειάζονται ή τερματίζουν.
- Να διαχειρίζεται την εναλλαγή (Swapping) των διεργασιών μεταξύ κύριας μνήμης και δίσκου.



Σημαντικότητα της Διαχείρισης Μνήμης

Είναι απαραίτητη διότι:

- Η μνήμη είναι ένας ανεπαρκής πόρος και είναι απαραίτητη η αποτελεσματική χρήση της.
- Παρέχει ευκολία στον προγραμματισμό.
- Ενισχύει τον πολυπρογραμματισμό.
- Παρέχει ασφάλεια και προστασία στις εκτελούμενες διεργασίες.
- Οι προγραμματιστές επιζητούν την ελαχιστοποίηση του χρόνου προσπέλασης και τη μεγιστοποίηση του μεγέθους της μνήμης για την εκτέλεση των προγραμμάτων (ωστόσο η μνήμη για την εκτέλεση προγραμμάτων πρέπει να είναι cost-effective).



Απαιτήσεις Μνήμης Microsoft Windows

Λειτουργικό Σύστημα	Ημερομηνία Έκδοσης	Μικρότερη Απαίτηση Μνήμης	Συνιστώμενη Μνήμη
Windows 1.0	Νοέμβριος 1985	256KB	-
Windows 2.03	Νοέμβριος 1987	320KB	-
Windows 3.0	Μάρτιος 1990	896KB	1MB
Windows 3.1	Απρίλιος 1992	2.6KB	4MB
Windows 95	Αύγουστος 1995	8MB	16MB
Windows NT 4.0	Αύγουστος 1996	32MB	96MB
Windows 98	Ιούνιος 1998	24MB	64MB
Windows ME	Σεπτέμβριος 2000	32MB	128MB
Windows 2000 Professional	Φεβρουάριος 2000	64MB	128MB
Windows XP Home	Οκτώβριος 2001	64MB	128MB
Windows XP Professional	Οκτώβριος 2001	128MB	256MB
Windows Vista	Ιανουάριος 2007	512MB	1GB
Windows 7	Οκτώβριος 2009	512MB	1GB (32 bit) & 2GB (64 bit)
Windows 8	Οκτώβριος 2012	512MB	1GB (32 bit) & 2GB (64 bit)



Χαρακτηριστικά Διαχειριστών Μνήμης

Χαρακτηριστικά των διαχειριστών μνήμης

- Εκχωρούν την κύρια μνήμη σε διεργασίες.
- Αντιστοιχούν το χώρο διευθύνσεων της διεργασίας στην κύρια μνήμη.
- Ελαχιστοποιούν το χρόνο προσπέλασης χρησιμοποιώντας cost-effective τεχνικές, στατικές ή δυναμικές.
- Αλληλεπιδρούν με ειδικό hardware για τη διαχείριση της μνήμης (MMU) για να βελτιώσουν την απόδοση.



Λογική & Φυσική Διεύθυνση

- **Λογική διεύθυνση:** είναι η διεύθυνση μνήμης που γνωρίζει και χρησιμοποιεί η διεργασία. Ονομάζεται και ιδεατή διεύθυνση (virtual address).
- **Φυσική διεύθυνση:** είναι η διεύθυνση που αντιστοιχεί σε μια πραγματική διεύθυνση μνήμης (hardware).

Κατά το χρόνο μετάφρασης ή εκτέλεσης **η λογική διεύθυνση αντιστοιχίζεται σε πραγματική διεύθυνση.**



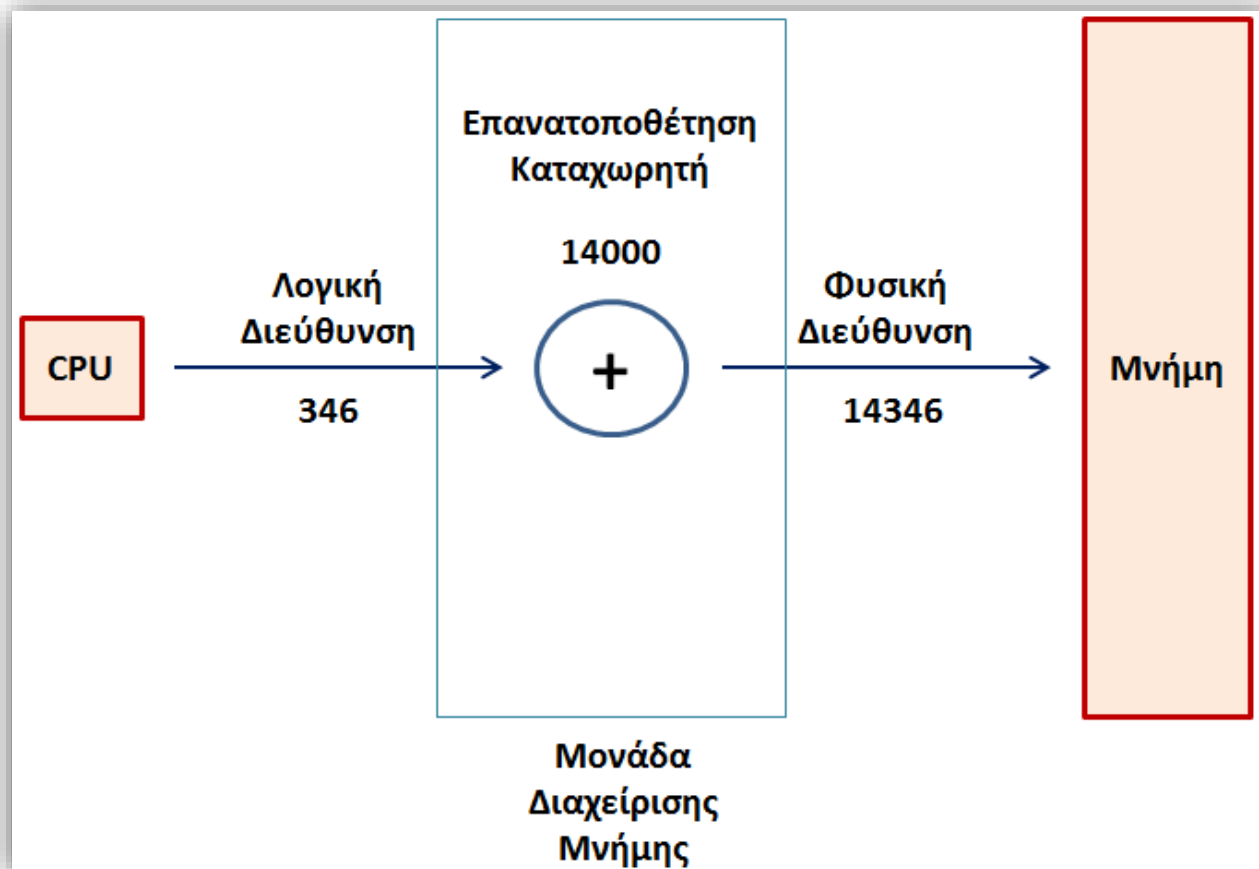
Μονάδα Διαχείρισης Μνήμης

- Φυσική συσκευή (**real hardware**) που χαρτογραφεί λογικές σε φυσικές διευθύνσεις.
- Σε ένα MMU σχήμα, η τιμή του καταχωρητή επανατοποθέτησης προστίθεται σε κάθε διεύθυνση που δημιουργείται από μία διεργασία χρήστη κατά τη χρονική στιγμή που αποστέλλεται στη μνήμη.
- Το πρόγραμμα χρήστη αφορά σε λογικές διευθύνσεις, δε χρειάζεται ποτέ να δει τις πραγματικές φυσικές διευθύνσεις.



Μονάδα Διαχείρισης Μνήμης

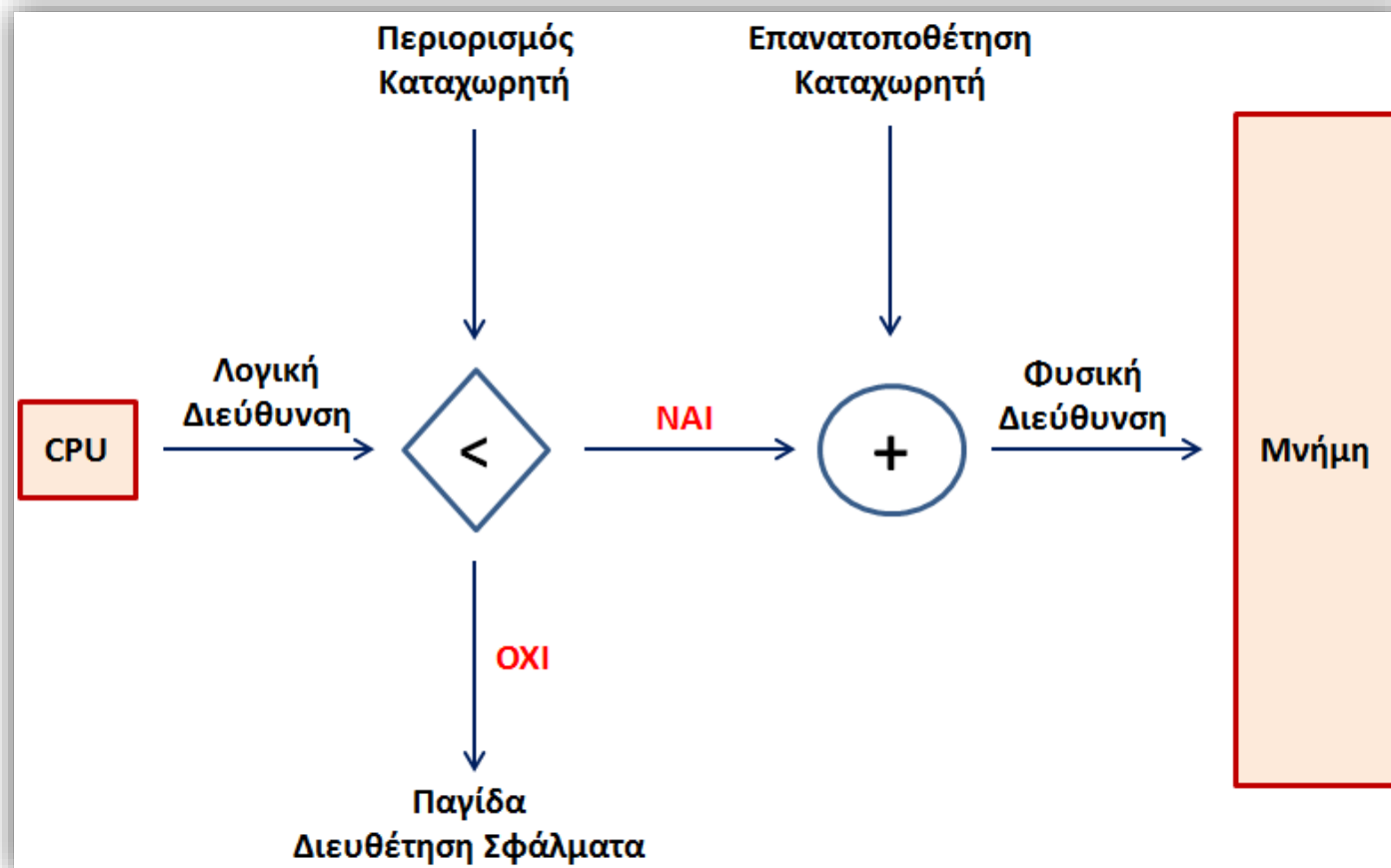
Παράδειγμα



Παράδειγμα Μονάδας Διαχείρισης Μνήμης

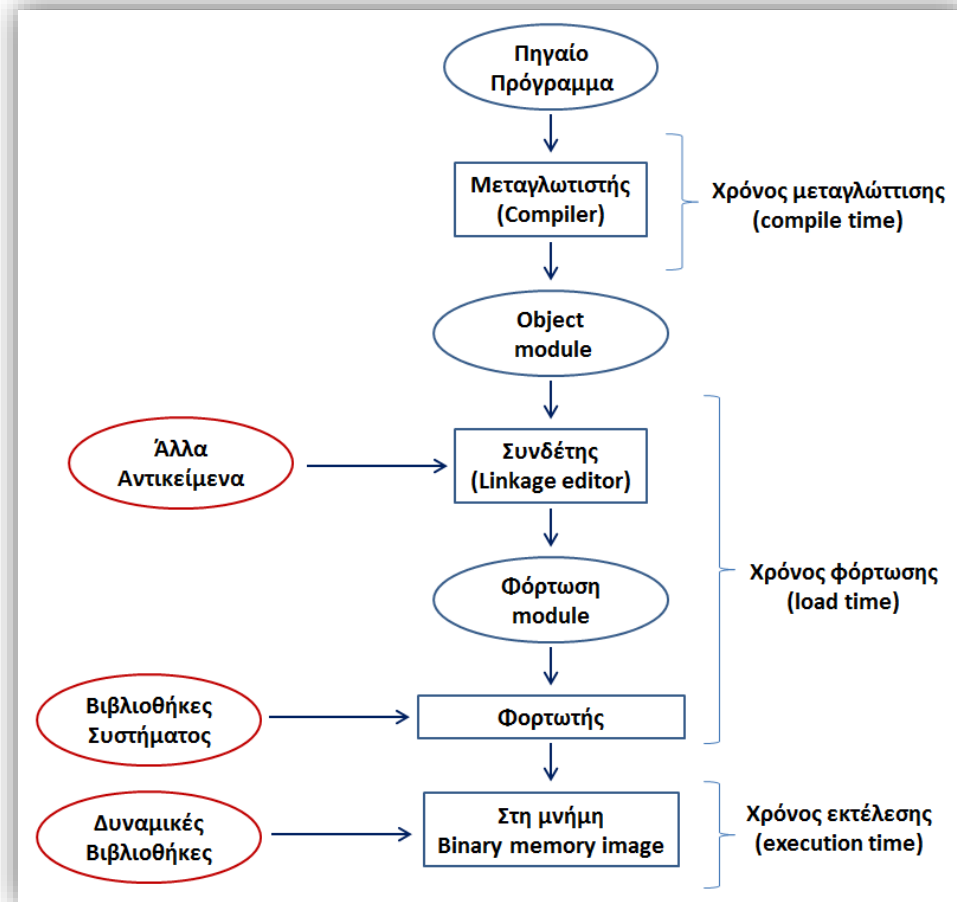


Μονάδα Διαχείρισης Μνήμης Με Προστασία



Παράδειγμα Μονάδας Διαχείρισης Μνήμης με προστασία

Δέσμευση Διευθύνσεων μίας διεργασίας



Δέσμευση διευθύνσεων μίας διεργασίας



Δέσμευση Διευθύνσεων

Δέσμευση διευθύνσεων για μια διεργασία γίνεται όταν:

- **Χρόνος μετάφρασης**

Εάν η θέση μνήμης είναι γνωστή εκ των προτέρων, μπορεί να δημιουργηθεί απόλυτος κώδικας. Πρέπει να επαναληφθεί η μετάφραση εάν αλλάξει η αρχική θέση.

- **Χρόνος φόρτωσης**

Εάν η θέση μνήμης δεν είναι γνωστή κατά το χρόνο μετάφρασης, πρέπει να δημιουργηθεί επανατοποθετήσιμος κώδικας. Ο τελικός συσχετισμός γίνεται στο χρόνο φόρτωσης.

- **Χρόνος εκτέλεσης**

Η δέσμευση καθυστερεί μέχρι τον χρόνο εκτέλεσης εάν η διεργασία μπορεί να μετακινηθεί κατά την εκτέλεση, από ένα τμήμα μνήμης σε κάποιο άλλο. Απαιτεί υποστήριξη από το υλικό για χαρτογράφηση των διευθύνσεων.



Εκχώρηση Μνήμης

Συνεχόμενη/Μη συνεχόμενη

- Η **συνεχόμενη εκχώρηση** μνήμης αφορά τα πρώτα υπολογιστικά συστήματα όπου αν το πρόγραμμα ήταν μεγαλύτερο από τη διαθέσιμη μνήμη το σύστημα δεν μπορούσε να το εκτελέσει.
- Στη **μη συνεχόμενη εκχώρηση** μνήμης το πρόγραμμα **διαιρείται σε τεμάχια** ή τμήματα που τοποθετούνται από το σύστημα σε όχι κατ' ανάγκη γειτονικές περιοχές στην κύρια μνήμη.

Η τεχνική αυτή κάνει εφικτή τη χρήση περιοχών που είναι πολύ μικρές για να χωρέσουν ολόκληρο πρόγραμμα. Αν και με τον τρόπο αυτό εισάγεται στο σύστημα πολυπλοκότητα αυτή δικαιολογείται από την αύξηση που επιτυγχάνεται στο βαθμό πολυπρογραμματισμού.



Βασική Διαχείριση Μνήμης

- **Μονοπρογραμματισμός**

Ένας χρήστης μονοπωλεί τη χρήση του συστήματος και όλοι οι πόροι είναι αφιερωμένοι σ' αυτόν.

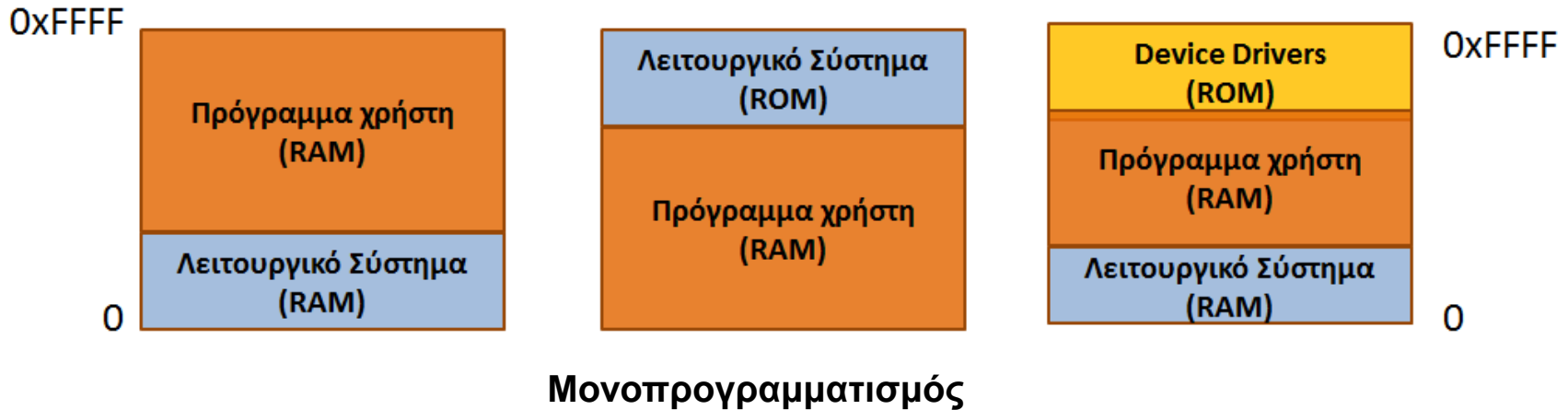
- **Επικαλύψεις (overlays)**

Τεχνική που επιτρέπει σε ένα σύστημα να εκτελεί προγράμματα που είναι μεγαλύτερα από την κύρια μνήμη.

Ο προγραμματιστής διαιρεί το πρόγραμμα σε λογικές ενότητες. Όταν το πρόγραμμα δε χρειάζεται μνήμη για ένα τμήμα, το σύστημα μπορεί να αντικαταστήσει όλη ή μέρη της κύριας μνήμης για να καλύψει μια ανάγκη (δηλ. να φορτώσει μια άλλη ενότητα).



Μονοπρογραμματισμός



- Τρεις απλοί τρόποι οργάνωσης μνήμης σε Λειτουργικό Σύστημα με μια διεργασία χρήστη.
- Η προστασία μνήμης δεν τίθεται ως πρόβλημα (Μόνο μια διεργασία υπάρχει).

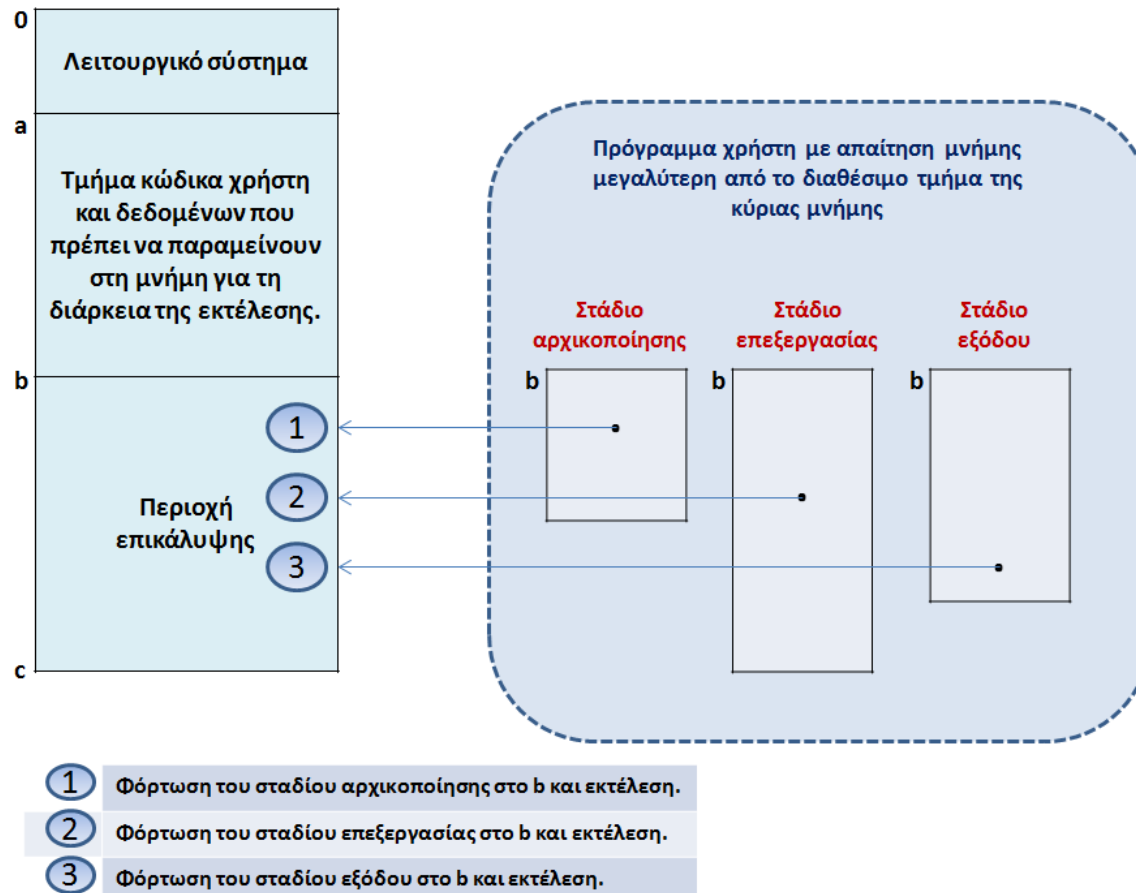


Επικαλύψεις/Διαστρωματώσεις

- Διατηρούμε στη μνήμη μόνο τις εντολές και τα δεδομένα εκείνα που απαιτούνται σε κάθε χρονική στιγμή.
- Χρειάζεται όταν η διεργασία είναι μεγαλύτερη από το χώρο μνήμης που της έχει ανατεθεί.
- Υλοποιείται από τον χρήστη, δεν χρειάζεται ιδιαίτερη υποστήριξη από το Λειτουργικό Σύστημα.



Επικαλύψεις (Overlays)



Επικαλύψεις

Πολυπρογραμματισμός και ποσοστό χρήσης CPU

- Η χρήση του επεξεργαστή από μία διεργασία διακόπτεται συχνά, λόγω της ανάγκης για λειτουργίες Εισόδου/Εξόδου (E/E) που είναι υπερβολικά αργές συγκρινόμενες με την ταχύτητα της Κεντρικής Μονάδας Επεξεργασίας (CPU).
- Η αύξηση της χρήσης της CPU επιτυγχάνεται με τα συστήματα πολυπρογραμματισμού, όπου αρκετοί χρήστες ανταγωνίζονται συγχρόνως για τους πόρους του συστήματος.
- Έτσι αρκετές διεργασίες πρέπει να βρίσκονται στην κύρια μνήμη την ίδια στιγμή, ώστε αν κάποια υλοποιεί λειτουργίες E/E κάποια άλλη να χρησιμοποιεί την CPU ώστε να αυξάνεται το ποσοστό χρήσης της CPU και η απόδοση (throughput) του συστήματος.



Χρήση CPU σε σύστημα ενός χρήστη

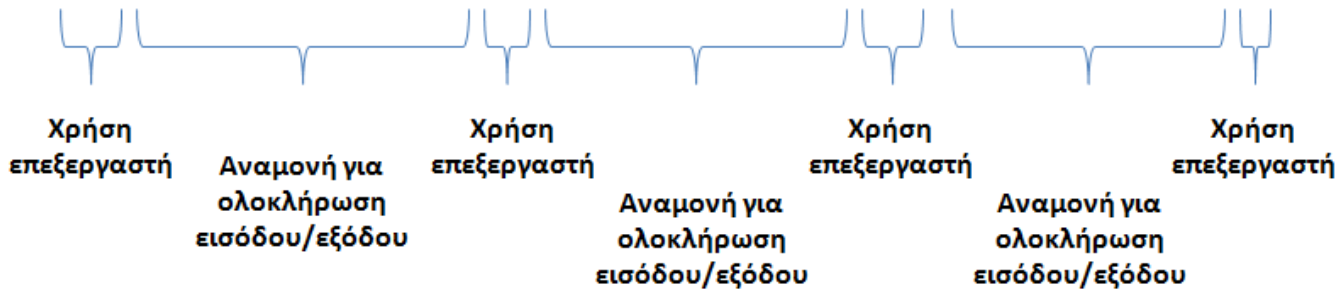
Μεγάλα IDLE Διαστήματα

Για διεργασία που πραγματοποιεί εντατικούς υπολογισμούς



Διαμοιραζόμενη περιοχή φανερώνει "Επεξεργαστής σε χρήση".

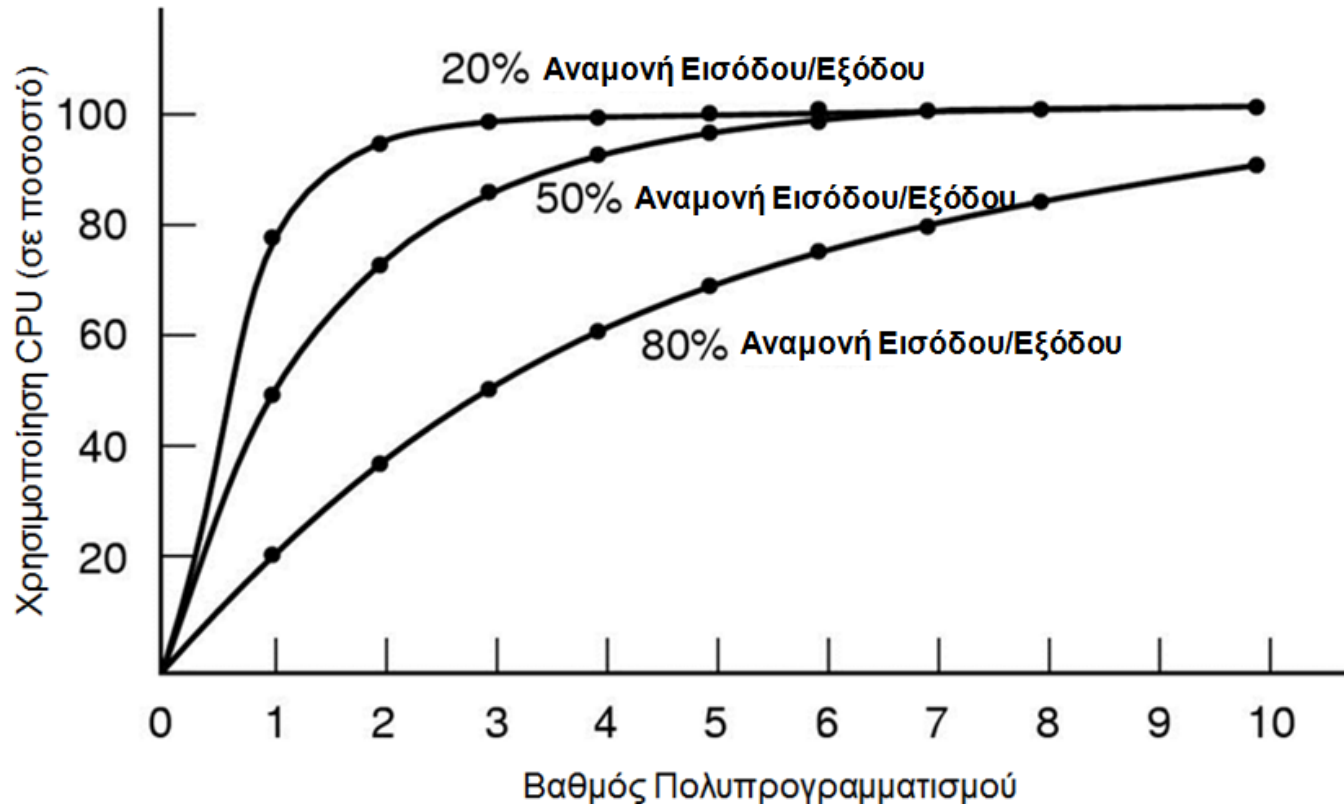
Για διεργασία που πραγματοποιεί τακτική είσοδο/έξοδο



Παράδειγμα χρήσης CPU σε σύστημα ενός χρήστη



Χρήση CPU ως συνάρτηση του πλήθους διεργασιών στη μνήμη



Χρήση CPU – Πλήθος Διεργασιών στη Μνήμη



Ικανοποιητικός αριθμός προγραμμάτων για μια CPU

Για τον καθορισμό του πλήθους των διεργασιών που μπορούν να υπάρχουν συγχρόνως στην κύρια μνήμη πρέπει να ληφθούν υπόψη και να εξισορροπηθούν τα εξής:

- Περισσότερες διεργασίες χρησιμοποιούν καλύτερα την CPU, αλλά απαιτείται καλύτερη διαχείριση και προστασία της μνήμης.
- Λιγότερες διεργασίες χρησιμοποιούν λιγότερη μνήμη (φθηνότερα).
- Περισσότερη αναμονή για E/E σημαίνει μικρότερη χρήση επεξεργαστή.



Προβλήματα του Πολυπρογραμματισμού

Η χρήση του πολυπρογραμματισμού εισάγει δύο θεμελιώδη προβλήματα:

- **Επανατοποθέτηση (relocation)**

Πιθανή τοποθέτηση διεργασιών σε διαφορετική διεύθυνση μνήμης κατά τη φόρτωση.

- **Προστασία (protection)**

Ανάγνωση και εγγραφή τμημάτων μνήμης που ανήκουν σε άλλες διεργασίες ή χρήστες.



Τμηματοποίηση σταθερού μεγέθους (Fixed Partitioning)

- **Ίσου μεγέθους τμήματα (equal-size partitions).**
 - Κάθε διεργασία με μέγεθος μικρότερο ή ίσο με το μέγεθος του τμήματος μπορεί να φορτωθεί στο διαθέσιμο τμήμα.
 - Αν όλα τα τμήματα είναι γεμάτα, το Λειτουργικό Σύστημα μπορεί να κάνει εναλλαγή μιας διεργασίας.
 - Ένα πρόγραμμα είναι πιθανό να μη χωρά σε ένα τμήμα. Ο προγραμματιστής πρέπει να σχεδιάσει το πρόγραμμα με επικαλύψεις.
- Με τη μέθοδο αυτή η χρησιμοποίηση της κύριας μνήμης είναι εξαιρετικά αναποτελεσματική (inefficient). Κάθε πρόγραμμα, όσο μικρό και να είναι, καταλαμβάνει ένα ολόκληρο τμήμα.
- Ο ανεκμετάλλετος χώρος εσωτερικά σε ένα τμήμα αναφέρεται ως **εσωτερικός κατακερματισμός (internal fragmentation).**

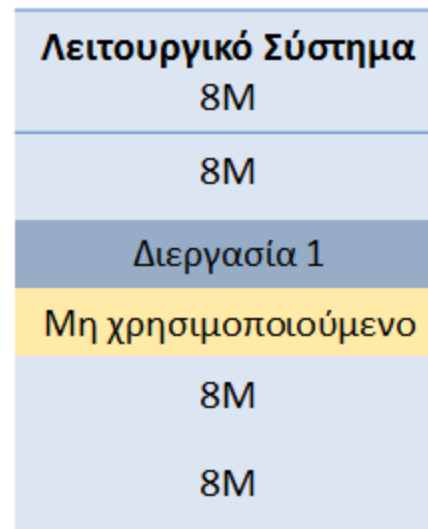
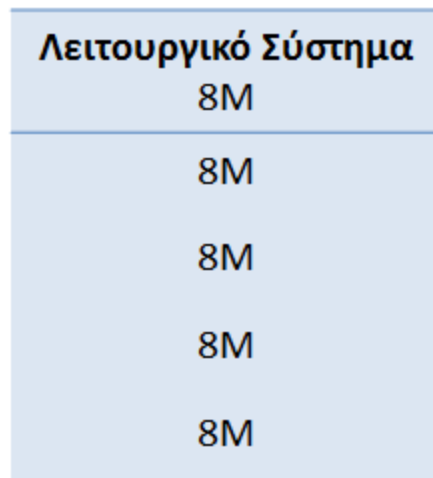


Τμηματοποίηση σταθερου μεγέθους

Εσωτερικός Κατακερματισμός

Εσωτερικός Κατακερματισμός

Μέρος του τμήματος που δε χρησιμοποιείται.



Εσωτερικός
Κατακερματισμός

Εσωτερικός Κατακερματισμός



Τμηματοποίηση σταθερου μεγέθους Πλεονεκτήματα/Μειονεκτήματα

Πλεονεκτήματα

- Μικρή επιβάρυνση στο Λειτουργικό Σύστημα.

Μειονεκτήματα

- Ανεπαρκής χρήση της μνήμης λόγω του εσωτερικού κατακερματισμού που κρατά σταθερό τον μέγιστο αριθμό διεργασιών που μπορούν να εκτελεστούν.
- Οι μικρές διεργασίες δε χρησιμοποιούν αποτελεσματικά το χώρο των τμημάτων.



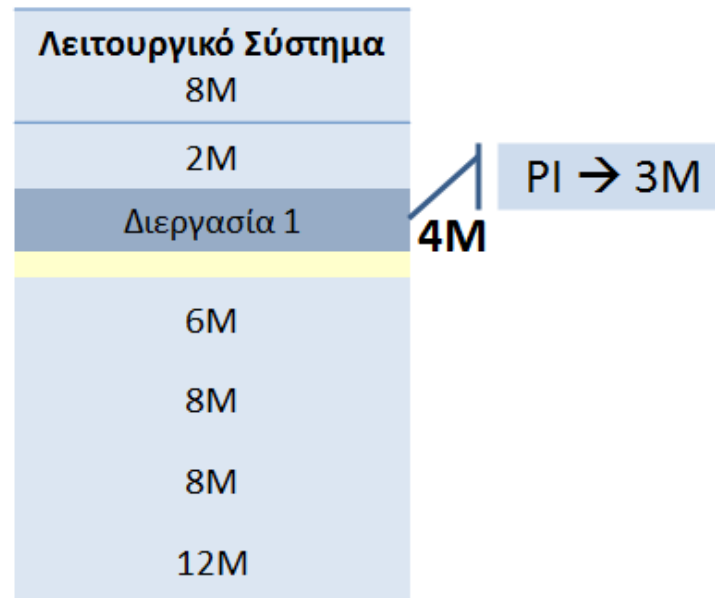
Τμηματοποίηση σταθερου μεγέθους

Άνισα Τμήματα

Μειώνει τα προβλήματα της τμηματοποίησης ίσων τμημάτων.

Πλεονεκτήματα

Μειώνεται ο εσωτερικός κατακερματισμός. Περισσότερο αποτελεσματική χρήση της κύριας μνήμης, σε σχέση με τη μέθοδο των ίσων τμημάτων.



Άνισα Τμήματα



Αλγόριθμοι Τοποθέτησης

Ίσου μεγέθους τμήματα

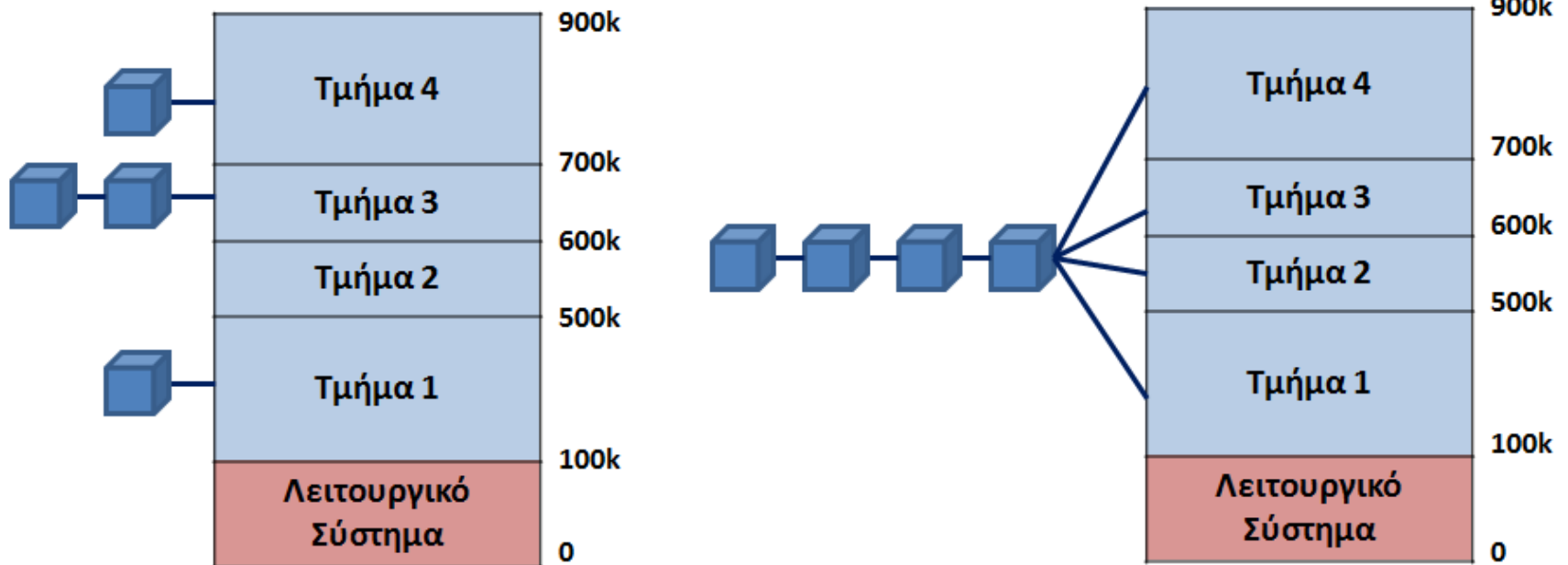
- Επειδή όλα τα τμήματα είναι ίσου μεγέθους, δεν έχει σημασία ποιο χρησιμοποιείται.
- Αν είναι όλα κατειλημμένα γίνεται εναλλαγή (*swapping*).

Διαφορετικού μεγέθους τμήματα

- **Ουρά για κάθε τμήμα**
 - Κάθε διεργασία μπορεί να αντιστοιχηθεί στο μικρότερο τμήμα στο οποίο χωρά.
 - Οι διεργασίες αντιστοιχούνται με τρόπο ώστε να ελαχιστοποιείται η σπατάλη μνήμης μέσα σε ένα τμήμα (μείωση του εσωτερικού κατακερματισμού).
- **Μια μοναδική ουρά για όλες τις διεργασίες**
 - Όταν η διεργασία πρέπει να φορτωθεί στη μνήμη επιλέγεται το μικρότερο διαθέσιμο τμήμα.
 - Καλύτερη ικανότητα για τη βελτιστοποίηση χρήσης της CPU.



Χρήση μοναδικής ή πολλαπλής ουράς



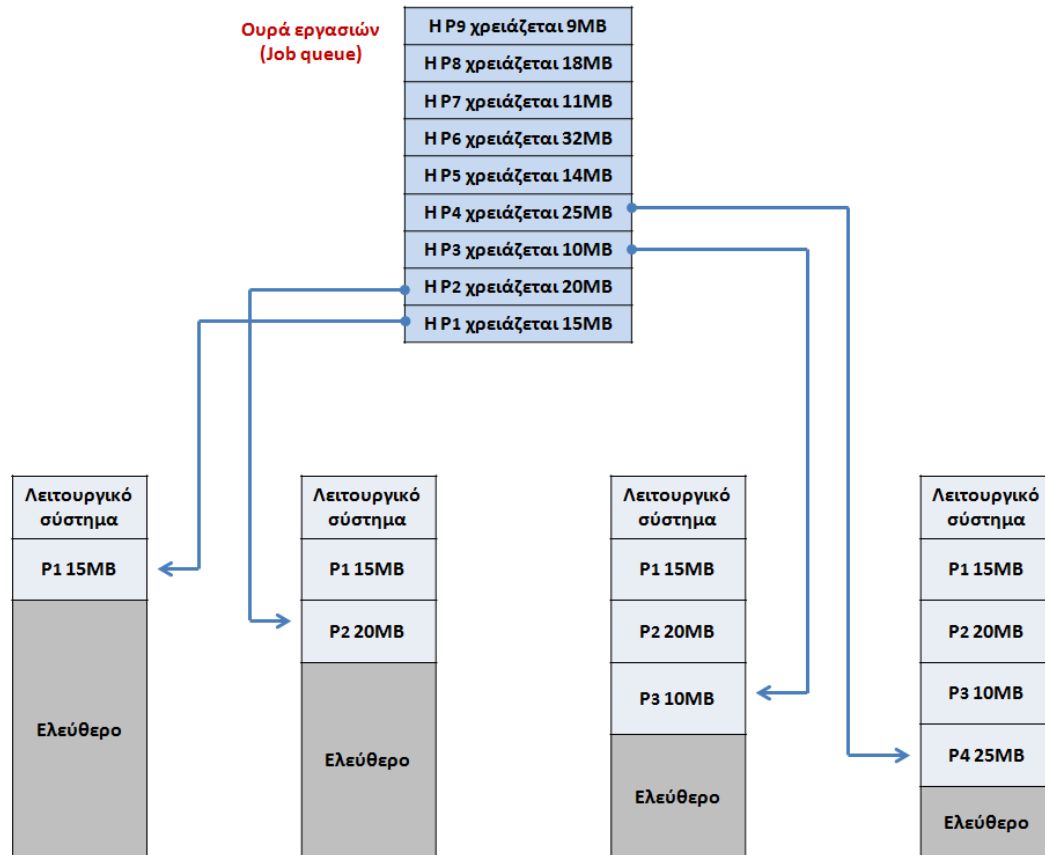
Παράδειγμα χρήσης μοναδικής ή πολλαπλής ουράς

Δυναμική τμηματοποίηση

- Τμήματα μεταβλητού μεγέθους και πλήθους.
- Μια διεργασία αντιστοιχείται ακριβώς στην ποσότητα μνήμης που απαιτείται.
- Τελικά υπάρχουν κενά στη μνήμη. Αυτός είναι ο **εξωτερικός κατακερματισμός (external fragmentation)**.
- Πρέπει να χρησιμοποιηθεί συμπίεση (**compaction**) που θα μετατοπίσει τις διεργασίες έτσι ώστε να είναι συνεχόμενες και όλη η ελεύθερη μνήμη να αποτελεί μια ενότητα (**block**).
- Η συμπίεση σπαταλά το χρόνο της CPU και προϋποθέτει τη δυνατότητα δυναμικής μετατόπισης (μεταφορά ενός προγράμματος σε άλλη περιοχή μνήμης χωρίς να ακυρώνονται οι αναφορές της μνήμης).



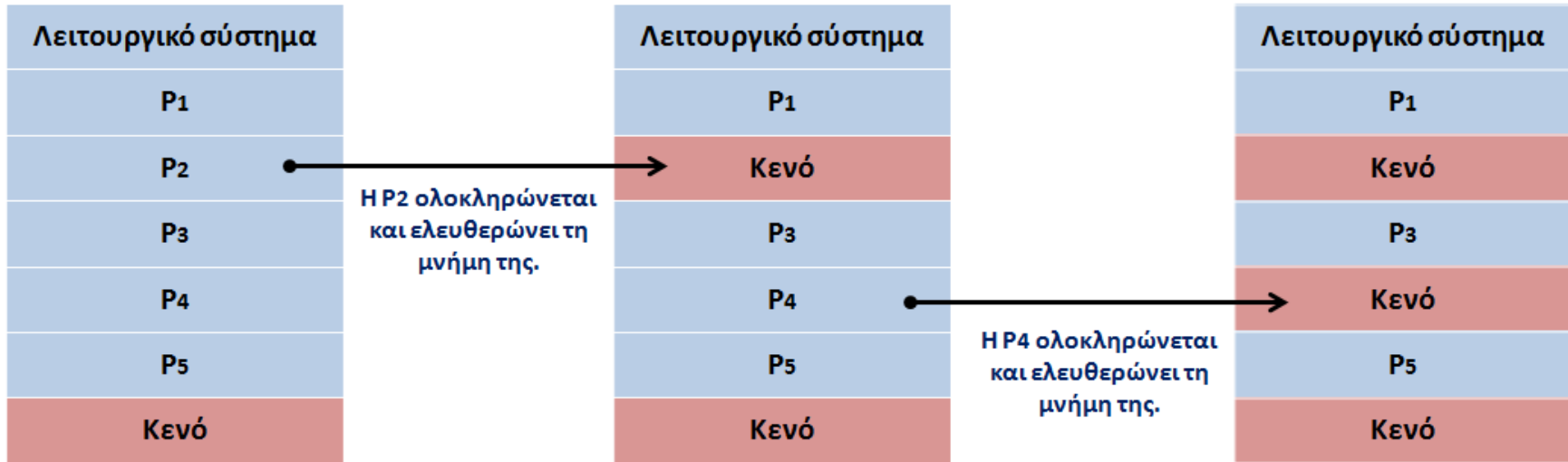
Δυναμική τμηματοποίηση (Παράδειγμα 1)



Παράδειγμα δυναμικής τμηματοποίησης



Δυναμική τμηματοποίηση (Παράδειγμα 2)



Παράδειγμα δυναμικής τμηματοποίησης



Εσωτερικός/Εξωτερικός Κατακερματισμός

Εσωτερικός

Είναι η μνήμη που δε χρησιμοποιείται (δαπανάται) και είναι ορατή μόνο από τη διεργασία που ζητά μνήμη. Συμβαίνει επειδή η ποσότητα μνήμης που θα εκχωρηθεί στη διεργασία πρέπει να είναι μεγαλύτερη ή ίση από την αιτούμενη ποσότητα.

Εξωτερικός

Είναι η μνήμη που δε χρησιμοποιείται (δαπανάται) και είναι ορατή από το σύστημα εκτός των διεργασιών που απαιτούν μνήμη. Συμβαίνει επειδή όλες οι απαιτήσεις μνήμης δεν είναι του ίδιου μεγέθους.



Αλγόριθμοι τοποθέτησης δυναμικής τμηματοποίησης (1/4)

Το Λ.Σ. πρέπει να αποφασίσει ποιο ελεύθερο τμήμα της μνήμης θα εκχωρήσει σε μια διεργασία.

Αλγόριθμος καλύτερης τοποθέτησης (best-fit algorithm)

- ▶ Επιλογή του block που είναι **πλησιέστερα** στο μέγεθος που απαιτείται.
- ▶ Έχει τη χειρότερη απόδοση.
- ▶ Μια και βρίσκει το μικρότερο block για τη διεργασία, η κύρια μνήμη γεμίζει γρήγορα από blocks που είναι πολύ μικρά και η συμπίεση πραγματοποιείται πιο συχνά.



Αλγόριθμοι τοποθέτησης δυναμικής τμηματοποίησης (2/4)

Αλγόριθμος πρώτης τοποθέτησης (First-fit algorithm)

- Ξεκινά και σαρώνει τη μνήμη από την αρχή και επιλέγει το πρώτο διαθέσιμο μπλοκ που είναι αρκετά μεγάλο.
- Ταχύτερος.
- Πολλές διεργασίες φορτώνονται στο εμπρός τμήμα της μνήμης που θα πρέπει να εξετάζεται κάθε φορά που γίνεται προσπάθεια για την εύρεση ενός ελεύθερου block.



Αλγόριθμοι τοποθέτησης δυναμικής τμηματοποίησης (3/4)

Αλγόριθμος επόμενης τοποθέτησης (next-fit algorithm)

- Ξεκινά να σαρώνει τη μνήμη από την τελευταία τοποθέτηση και επιλέγει το επόμενο αρκετά μεγάλο διαθέσιμο μπλοκ.
- Εκχωρεί συχνά ένα block μνήμης που βρίσκεται στο τέλος της μνήμης, όπου βρίσκεται το μεγαλύτερο block.
- Το μεγαλύτερο block μνήμης διασπάται σε μικρότερα blocks.
- Η συμπίεση απαιτείται για να αποκτηθεί ένα μεγάλο block στο τέλος της μνήμης.



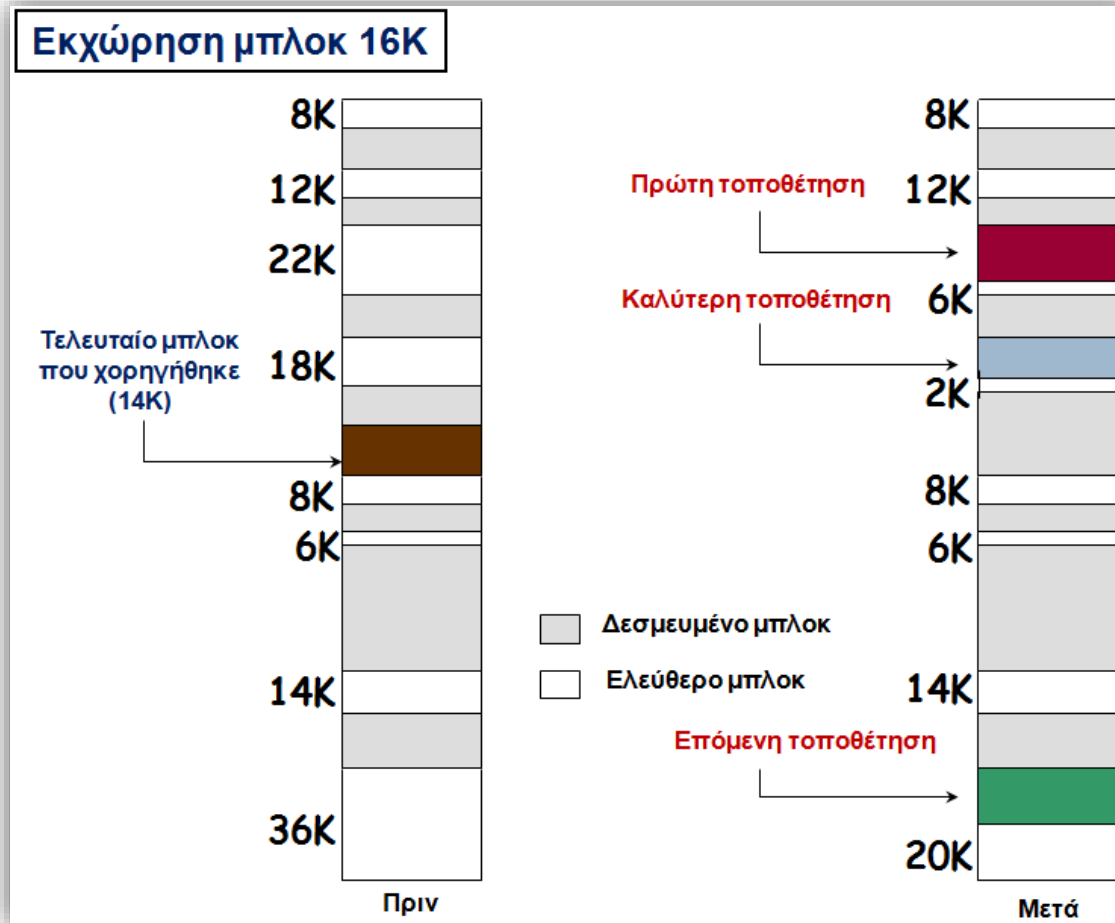
Αλγόριθμοι τοποθέτησης δυναμικής τμηματοποίησης (4/4)

Αλγόριθμος χειρότερης τοποθέτησης (worst-fit algorithm)

- Επιλέγει το μέγιστο διαθέσιμο μπλοκ που υπάρχει
- Αναζήτηση σε όλη τη μνήμη
- Καταλήγει στο μέγιστο δυνατό υπόλοιπο ανοιγμάτων.



Αλγόριθμοι τοποθέτησης (Παράδειγμα)



Παράδειγμα αλγορίθμου τοποθέτησης



Εναλλαγή, Ανταλλαγή (1/2)

- Μία διεργασία μπορεί να μεταφερθεί προσωρινά από τη μνήμη σε κάποιο υποστηρικτικό μέσο αποθήκευσης και στη συνέχεια να φορτωθεί ξανά στη μνήμη ώστε να συνεχίσει την εκτέλεση της.
- **Υποστηρικτικό μέσο αποθήκευσης (*backing store*)**
Γρήγορος δίσκος μεγάλης χωρητικότητας ώστε να χωράει όλα τα αντίγραφα των εικόνων όλων των χρηστών.
- Πρέπει να παρέχει απευθείας πρόσβαση σε αυτές τις εικόνες της μνήμης.
- **Κύλιση προς τα έξω/Κύλιση προς τα μέσα**
Παράμετρος ανταλλαγής που χρησιμοποιείται για αλγόριθμους δρομολόγησης που βασίζονται σε προτεραιότητες. Η διεργασία χαμηλότερης προτεραιότητας μετακινείται εκτός, ώστε να φορτωθεί και να εκτελεσθεί η διεργασία υψηλότερης προτεραιότητας.

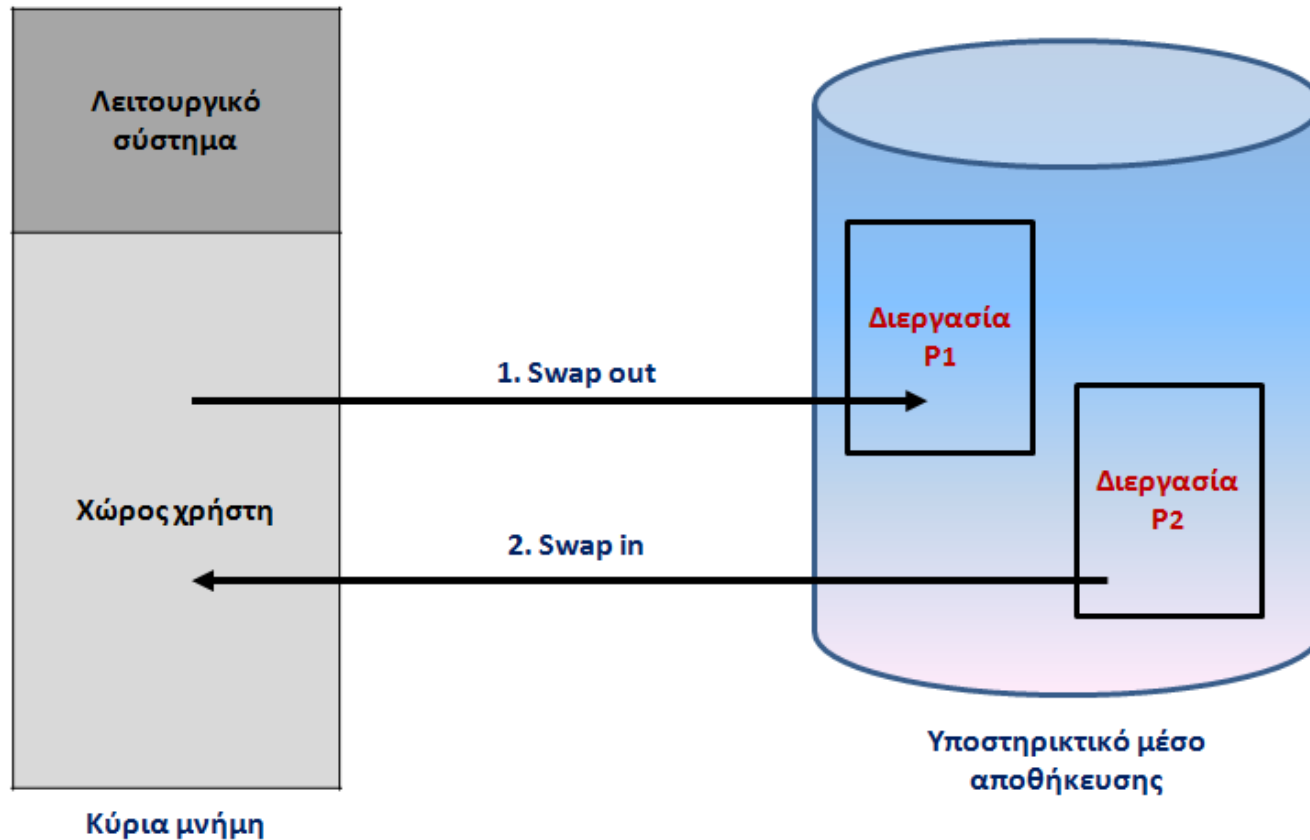


Εναλλαγή, Ανταλλαγή (2/2)

- Ο κυριότερος παράγοντας του χρόνου ανταλλαγής είναι ο **χρόνος μεταφοράς (transfer time)**. Ο συνολικός χρόνος μεταφοράς είναι ανάλογος του ποσού της μνήμης που μετακινείται και ανταλλάσσεται.
- Τροποποιημένες εκδοχές της ανταλλαγής υπάρχουν σε πολλά συστήματα, για παράδειγμα UNIX, Microsoft Windows.



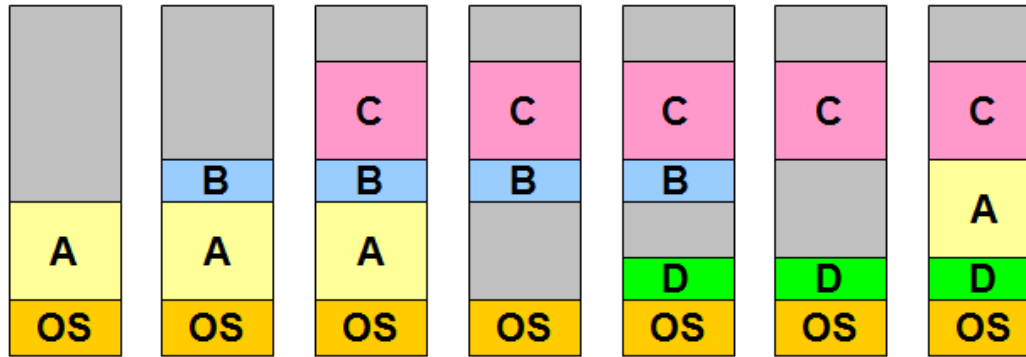
Εναλλαγή (Swapping) (1/2)



Παράδειγμα εναλλαγής



Εναλλαγή (Swapping) (2/2)



Η εκχώρηση μνήμης αλλάζει καθώς:

- Έρχονται νέες διεργασίες στη μνήμη.
- Διεργασίες εγκαταλείπουν τη μνήμη.
 - Εναλλάσσονται στο δίσκο.
 - Ολοκληρώνουν την εκτέλεσή τους.

Οι γκρι περιοχές είναι μνήμη που δε χρησιμοποιείται.



Εναλλαγή

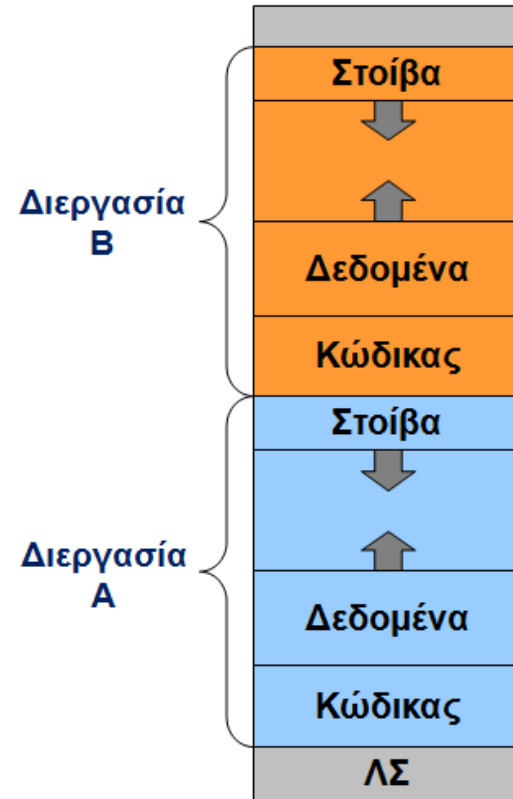
Αφήνοντας χώρο για ανάπτυξη

Ανάγκη για τη δυνατότητα ανάπτυξης των προγραμμάτων.

- Εκχώρηση περισσότερης μνήμης για δεδομένα.
- Μεγαλύτερη στοίβα (stack).

Εκχώρηση μεγαλύτερης ποσότητας μνήμης από όση απαιτείται καταρχήν.

- **Αναποτελεσματική:** σπατάλη μνήμης που δε χρησιμοποιείται.



Παράδειγμα εναλλαγής



Περιορισμοί της εναλλαγής

Προβλήματα με την εναλλαγή

- Η διεργασία πρέπει να χωρά στη φυσική μνήμη (αδύνατη η εκτέλεση μεγαλύτερων διεργασιών).
- Η μνήμη κατακερματίζεται (**fragmented**).

Οι επικαλύψεις επιλύουν το πρώτο πρόβλημα

- Κατανέμουν τη διεργασία στη διάρκεια του χρόνου (κυρίως τα δεδομένα).
- Δεν επιλύουν το πρόβλημα του κατακερματισμού.



Ασκήσεις (1/3)

Υποθέστε ότι έχετε ελεύθερη μνήμη σε τμήματα μεγέθους 100KB, 500KB, 200KB, 300KB, and 600KB (με αυτή τη σειρά) και υπάρχουν κατά σειρά απαιτήσεις μνήμης για 212KB, 417KB, 112KB, and 426KB.

- Δείξτε πως ο αλγόριθμος **πρώτης τοποθέτησης** θα διευθετήσει τις απαιτήσεις στην ελεύθερη μνήμη. Υπολογίστε το ελεύθερο τμήμα μνήμης μετά από κάθε άφιξη.
- Δείξτε πως ο αλγόριθμος **βέλτιστης τοποθέτησης** θα διευθετήσει τις ίδιες απαιτήσεις στην ελεύθερη μνήμη. Υπολογίστε το ελεύθερο τμήμα μνήμης μετά από κάθε άφιξη.
- Δείξτε πως ο αλγόριθμος **επόμενης τοποθέτησης** θα διευθετήσει τις απαιτήσεις στην ελεύθερη μνήμη (last allocated block 200KB). Υπολογίστε το ελεύθερο τμήμα μνήμης μετά από κάθε άφιξη.



Ασκήσεις (2/3)

Ένα σύστημα τοποθετεί διεργασίες στη μνήμη χρησιμοποιώντας δυναμική πολιτική τοποθέτησης. Κατά την πλέον πρόσφατη χρονική στιγμή έγινε φόρτωση μιας διεργασίας που χρειαζόταν 12KB μνήμης και η εικόνα μνήμης του συστήματος διαμορφώθηκε ως εξής:



Οι σκιασμένες περιοχές δηλώνουν χρησιμοποιημένα τμήματα μνήμης, οι λευκές τα αχρησιμοποίητα τμήματα ενώ η περιοχή με μαύρο χρώμα τη θέση όπου έγινε η τελευταία τοποθέτηση. Οι αριθμοί δηλώνουν το μέγεθος σε KB.

- Να σχεδιάσετε την εικόνα μνήμης μετά την τοποθέτηση μιας νέας διεργασίας που χρειάζεται 22KB μνήμης σύμφωνα με τους αλγορίθμους : first-fit, best-fit, next-fit.



Ασκήσεις (3/3)

Ο διαχειριστής μνήμης ενός συστήματος που χρησιμοποιεί στρατηγική τμημάτων μεταβλητού μεγέθους διαθέτει ελεύθερα τμήματα μεγέθους 600, 400, 1000, 2200, 1600, 1050 bytes. Θεωρήστε κατά σειρά τις παρακάτω διαδοχικές απαιτήσεις:

- Ποιο τμήμα θα επιλεγεί για απαίτηση 1603 bytes με τη μέθοδο best-fit;
- Ποιο τμήμα θα επιλεγεί για απαίτηση 949 bytes με τη μέθοδο best-fit;
- Ποιο τμήμα θα επιλεγεί για απαίτηση 963 bytes με τη μέθοδο first-fit;
- Ποιο τμήμα θα επιλεγεί για απαίτηση 349 bytes με τη μέθοδο first-fit;
- Υποθέστε ότι η λίστα ελεύθερων τμημάτων διατάσσεται κατά αύξουσα σειρά μεγέθους των τμημάτων που αναφέρονται στην αρχή της άσκησης. Ποιο τμήμα θα επιλεγεί για απαίτηση 1603 bytes με τη μέθοδο first-fit;



Σελιδοποίηση

- Κατάτμηση της μνήμης σε μικρά τμήματα ίσου μεγέθους και χωρισμός κάθε διεργασίας σε τμήματα του ίδιου μεγέθους.
- Τα τμήματα της διεργασίας λέγονται σελίδες (**pages**).
- Τα τμήματα της μνήμης λέγονται πλαίσια (**frames**).
- Το Λειτουργικό Σύστημα διατηρεί έναν πίνακα διεργασιών (**page table**) για κάθε διεργασία.
 - Περιέχει τη θέση του πλαισίου για κάθε σελίδα της διεργασίας.
 - Η διεύθυνση μνήμης αποτελείται από τον αριθμό της σελίδας (**page number**) και τη μετατόπιση (**offset**) μέσα στη σελίδα.



Διεργασία και Πλαίσια

Αριθμός πλαisiού	Κύρια μνήμη
0	A.0
1	A.1
2	A.2
3	A.3
4	D.0
5	D.1
6	D.2
7	C.0
8	C.1
9	C.2
10	C.3
11	D.3
12	D.4
13	
14	

Παράδειγμα εναλλαγής με πλαίσια



Πίνακας Σελίδας

0	0
1	1
2	2
3	3

Διεργασία A
Πίνακας σελίδας

0	-
1	-
2	-

Διεργασία B
Πίνακας σελίδας

0	7
1	8
2	9
3	10

Διεργασία C
Πίνακας σελίδας

0	4
1	5
2	6
3	11
4	12

Διεργασία D
Πίνακας σελίδας

13
14

Ελεύθερο
πλαίσιο λίστας

Παράδειγμα δομών δεδομένων



Κατάτμηση (Segmentation)

Ένα πρόγραμμα μπορεί να διαιρεθεί σε τμήματα:

- Τα τμήματα μπορούν να είναι διαφορετικών μεγεθών.
- Υπάρχει ένα μέγιστο μήκος τμήματος.

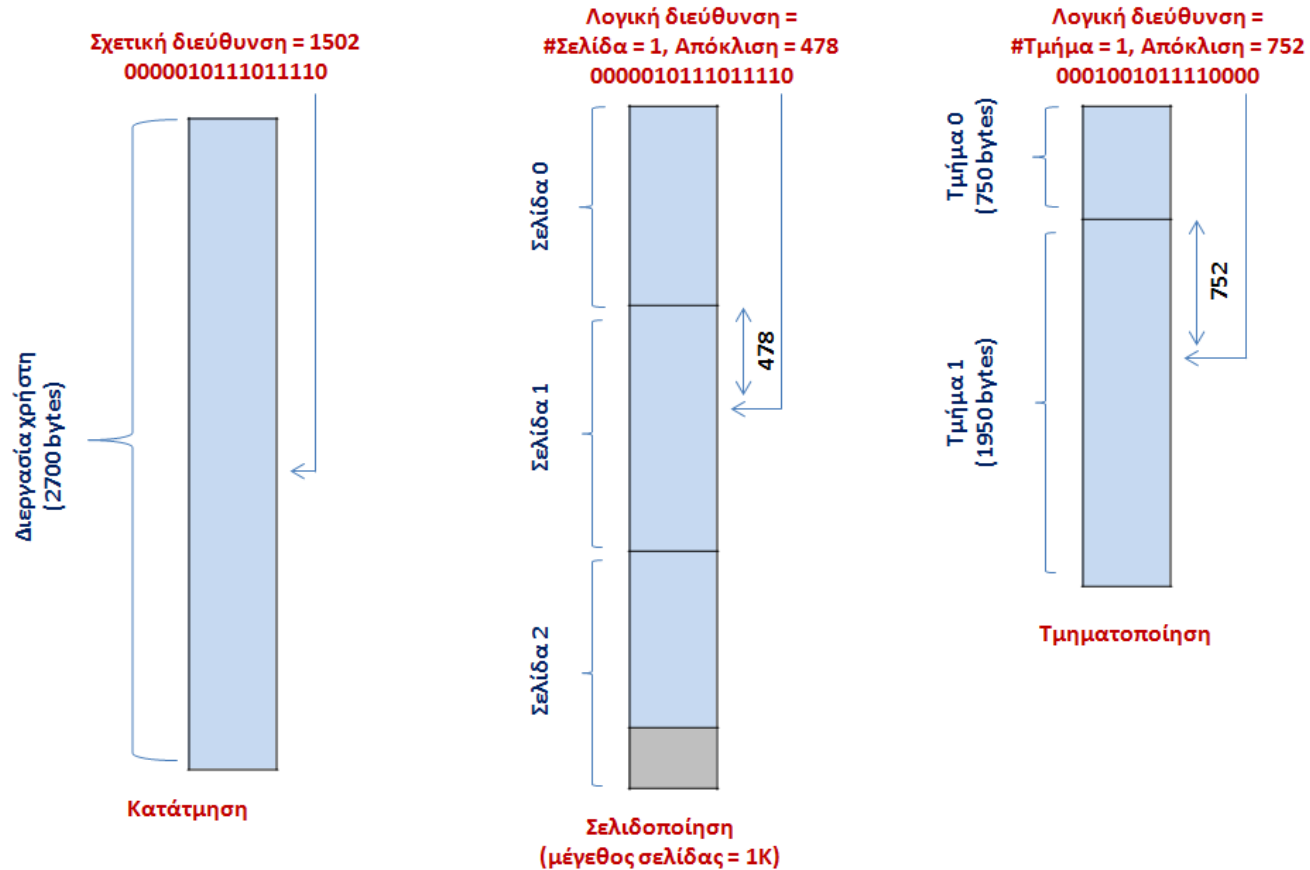
Η διεύθυνση αποτελείται από 2 μέρη:

- Τον αριθμό του τμήματος (**segment number**).
- Τη μετατόπιση (**offset**).

Η κατάτμηση είναι όμοια με τη δυναμική τμηματοποίηση.



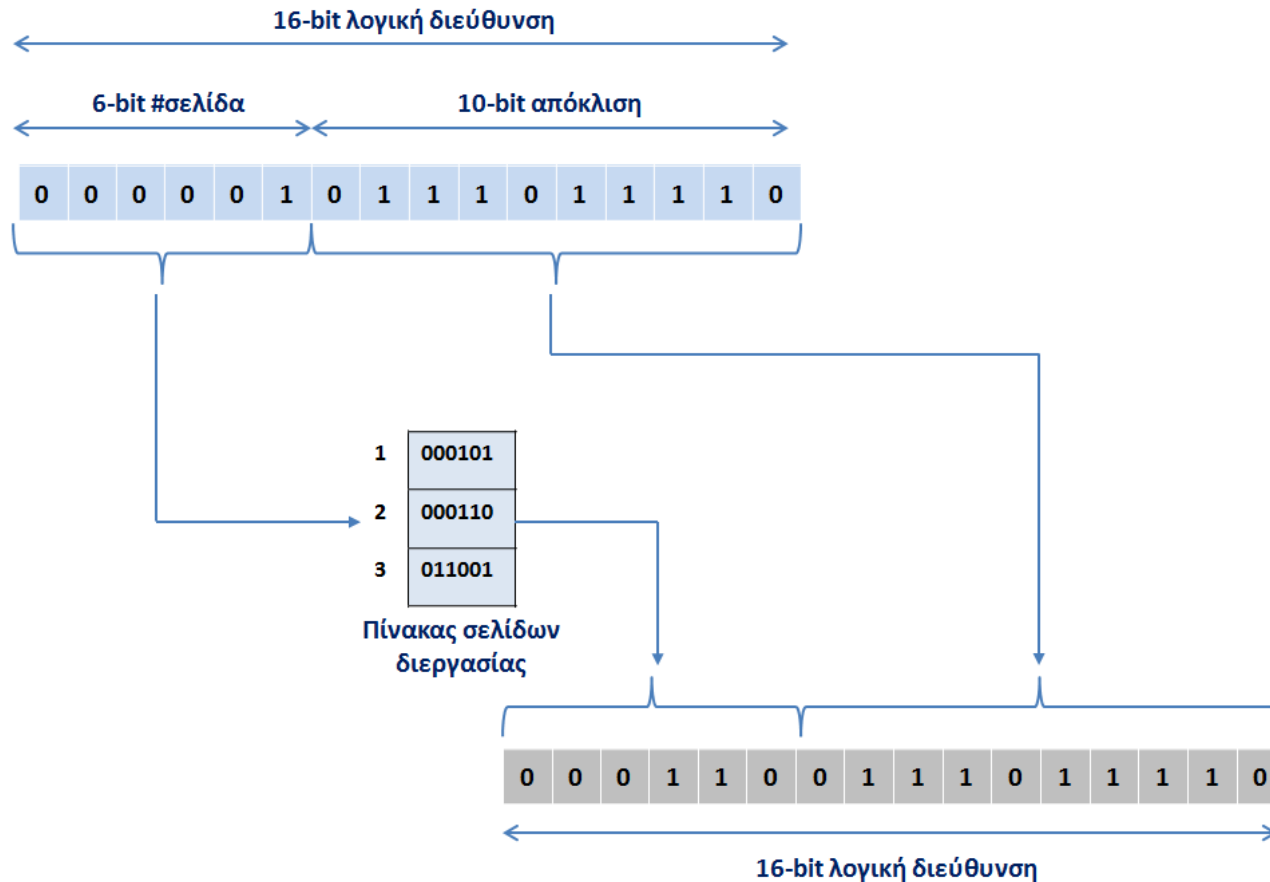
Λογικές Διευθύνσεις



Παράδειγμα λογικών διευθύνσεων



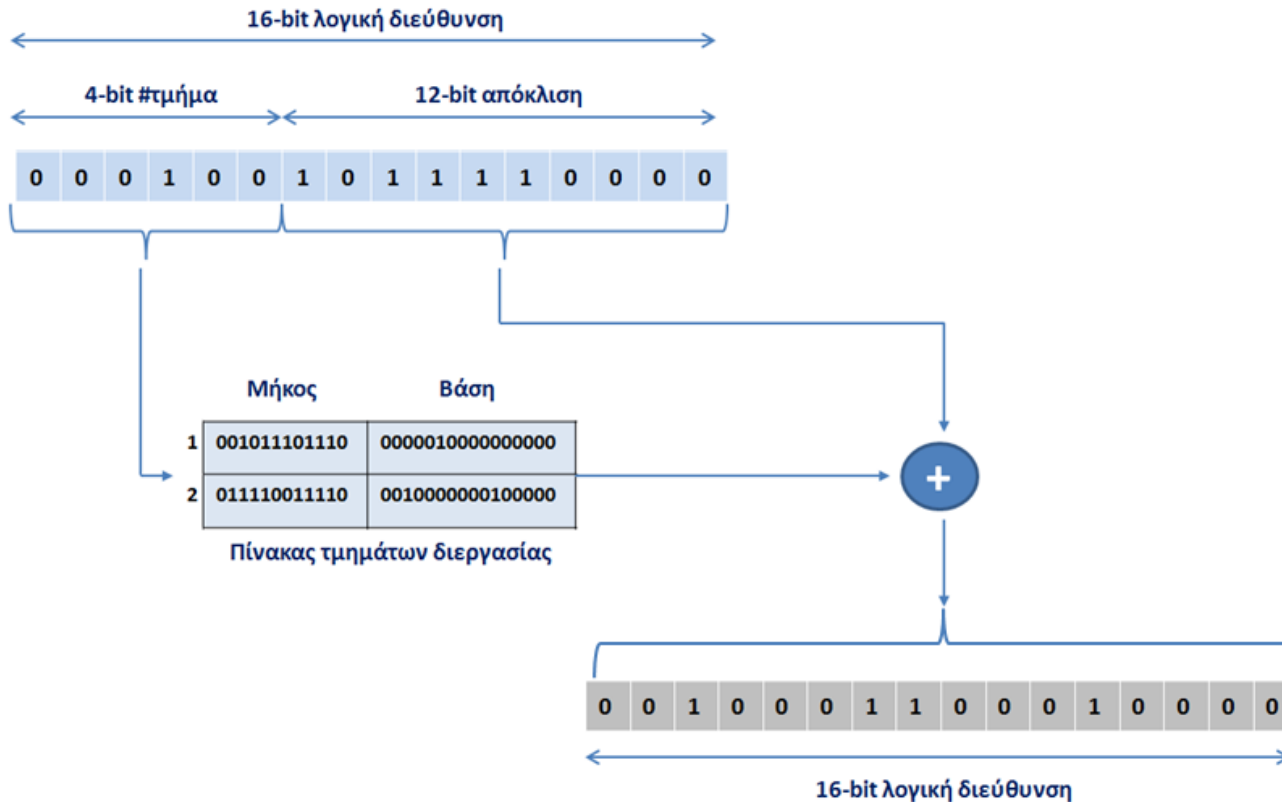
Σελιδοποίηση (Παράδειγμα)



Παράδειγμα σελιδοποίησης



Κατάτμηση (Παράδειγμα)



Παράδειγμα κατάτμησης

Σύγκριση (1/2)

Μέθοδος	Περιγραφή	Πλεονεκτήματα	Μειονεκτήματα
Τμηματοποίηση Σταθερού Μεγέθους	Η κύρια μνήμη χωρίζεται σε έναν αριθμό στατικών τμημάτων. Μια διεργασία μπορεί να φορτωθεί σε ένα τμήμα ίσου ή μεγαλύτερου μεγέθους.	Ευκολία στην υλοποίηση, μικρό κόστος για το Λειτουργικό Σύστημα.	Ανεπαρκής χρήση της μνήμης λόγω εσωτερικού κατακερματισμού, ο μέγιστος αριθμός των ενεργών διεργασιών είναι προκαθορισμένος.
Δυναμική Τμηματοποίηση	Τα τμήματα δημιουργούνται δυναμικά έτσι ώστε κάθε διεργασία να φορτώνεται σε τμήμα μεγέθους ίσο με το δικό της.	Δεν υπάρχει εσωτερικός κατακερματισμός, αποτελεσματικότερη χρησιμοποίηση της κύριας μνήμης.	Μη αποτελεσματική χρήση της CPU εξαιτίας της ανάγκης για συμπίεση για την αντιμετώπιση του εξωτερικού κατακερματισμού.



Σύγκριση (2/2)

Μέθοδος	Περιγραφή	Πλεονεκτήματα	Μειονεκτήματα
Σελιδοποίηση	Η κύρια μνήμη χωρίζεται σε έναν αριθμό ίδιου μεγέθους πλαισίων. Κάθε διεργασία χωρίζεται σε έναν αριθμό σελίδων ίδιου μεγέθους με τα πλαίσια. Μια διεργασία φορτώνεται με τη φόρτωση όλων των σελίδων της στα διαθέσιμα, όχι απαραίτητως συνεχόμενα πλαίσια.	Δεν υπάρχει εξωτερικός κατακερματισμός.	Μικρό ποσό εσωτερικού κατακερματισμού.
Κατάτμηση	Κάθε διεργασία χωρίζεται σε έναν συγκεκριμένο αριθμό από segments. Η διεργασία φορτώνεται φορτώνοντας όλα της τα segments σε δυναμικά τμήματα που δεν χρειάζεται να είναι συνεχόμενα.	Δεν υπάρχει εσωτερικός κατακερματισμός, βελτιωμένη χρήση της μνήμης και μειωμένο κόστος σε σχέση με τη δυναμική τμηματοποίηση.	Εξωτερικός κατακερματισμός.



Αναφορές

- [1]. Stallings William, “Operating systems: Internal and Design Principles”, Fourth edition, Publishing as Prentice Hall, 2000.
- [2]. A.S.Tanenbaum, "Σύγχρονα Λειτουργικά Συστήματα" Τόμος Α', Εκδόσεις, Παπασωτηρίου.
- [3]. H.M. Deitel, "Operating Systems", 2nd edition, Addison-Wesley Publishing Company.
- [3]. W. Stallings, “Λειτουργικά Συστήματα Αρχές Σχεδίασης”, 6η έκδοση, ΕΚΔΟΣΕΙΣ ΤΖΙΟΛΑ, 2009, Θεσσαλονίκη.





Τέλος Ενότητας



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΣΠΑ
2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

