



Ελληνική Δημοκρατία
Τεχνολογικό Εκπαιδευτικό
Ίδρυμα Ηπείρου

Λειτουργικά Συστήματα

Ενότητα 8 : Διαχείριση Μνήμης

Δημήτριος Λιαροκάπης



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Τμήμα Μηχανικών Πληροφορικής Τ.Ε

Λειτουργικά Συστήματα

Ενότητα 8 : Διαχείριση Μνήμης

Δημήτριος Λιαροκάπης

Καθηγητής Εφαρμογών

Άρτα, 2015



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης





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 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.
- Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο ΤΕΙ Ηπείρου**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ



Διαχείριση Μνήμης

- Κύρια Μνήμη
 - Εισαγωγή
 - Βασική διαχείριση μνήμης
 - Μνήμη και πολυπρογραμματισμός
 - Τμηματοποίηση σταθερού μεγέθους
 - Δυναμική τμηματοποίηση
 - Εναλλαγή



Κύρια Μνήμη

- Εισαγωγή
- Βασική διαχείριση μνήμης
- Μνήμη και πολυπρογραμματισμός
- Τμηματοποίηση σταθερού μεγέθους
 - ✓ Ίσα τμήματα
 - ✓ Άνισα τμήματα
- Δυναμική τμηματοποίηση
- Εναλλαγή



1. Εισαγωγή

- 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.
- 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.



Διαχειριστής μνήμης

- Ο διαχειριστής μνήμης είναι ένα συστατικό του ΛΣ που ασχολείται με την οργάνωση και τις στρατηγικές διαχείρισης της μνήμης.
- Χαρακτηριστικά των διαχειριστών μνήμης:
 - Εκχωρούν την πρωτεύουσα 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.



Στρατηγικές διαχείρισης μνήμης

- Σχεδιάζονται έτσι ώστε να είναι εφικτή η βέλτιστη δυνατή χρήση της κύριας μνήμης και διακρίνονται σε :
 - Στρατηγικές προσκόμισης
 - Καθορίζουν το σημείο όπου θα τοποθετηθεί το επόμενο τμήμα προγράμματος ή δεδομένων, καθώς μετακινείται από τη δευτερεύουσα μνήμη.
 - Στρατηγικές τοποθέτησης
 - Καθορίζουν το σημείο της κύριας μνήμης όπου το σύστημα θα μπορούσε να τοποθετήσει τμήματα δεδομένων.
 - Στρατηγικές επανατοποθέτησης
 - Καθορίζουν ποιο τμήμα θα αφαιρεθεί από την κύρια μνήμη στις περιπτώσεις όπου η κύρια μνήμη είναι αρκετά πλήρης ώστε να παρέχει χώρο σε ένα νέο πρόγραμμα.



Εκχώρηση μνήμης

- Συνεχόμενη και μη συνεχόμενη εκχώρηση μνήμης
- Συνεχόμενη εκχώρηση μνήμης
 - Ολόκληρο το πρόγραμμα τοποθετείται σε γειτονικές θέσεις μνήμης
 - Αφορά τα πρώτα υπολογιστικά συστήματα όπου αν το πρόγραμμα ήταν μεγαλύτερο από τη διαθέσιμη μνήμη το σύστημα δεν μπορούσε να το εκτελέσει.
- Μη συνεχόμενη εκχώρηση μνήμης
 - Το πρόγραμμα διαιρείται σε τεμάχια ή τμήματα που τοποθετούνται από το σύστημα σε μη γειτονικές σχισμές 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.
 - Εφικτή μέσω της εικονικής μνήμης

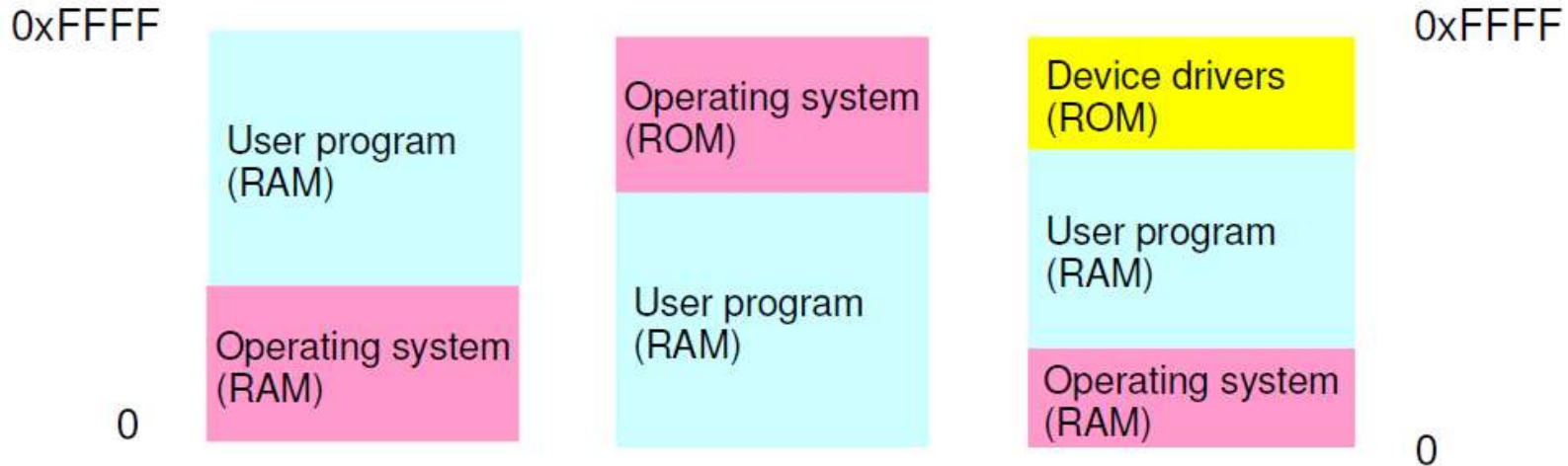


2. Βασική διαχείριση μνήμης

- 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 (ή το τμήμα μνήμης).
 - Ο προγραμματιστής διαιρεί το πρόγραμμα σε λογικές ενότητες.
 - Όταν το πρόγραμμα δεν 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)



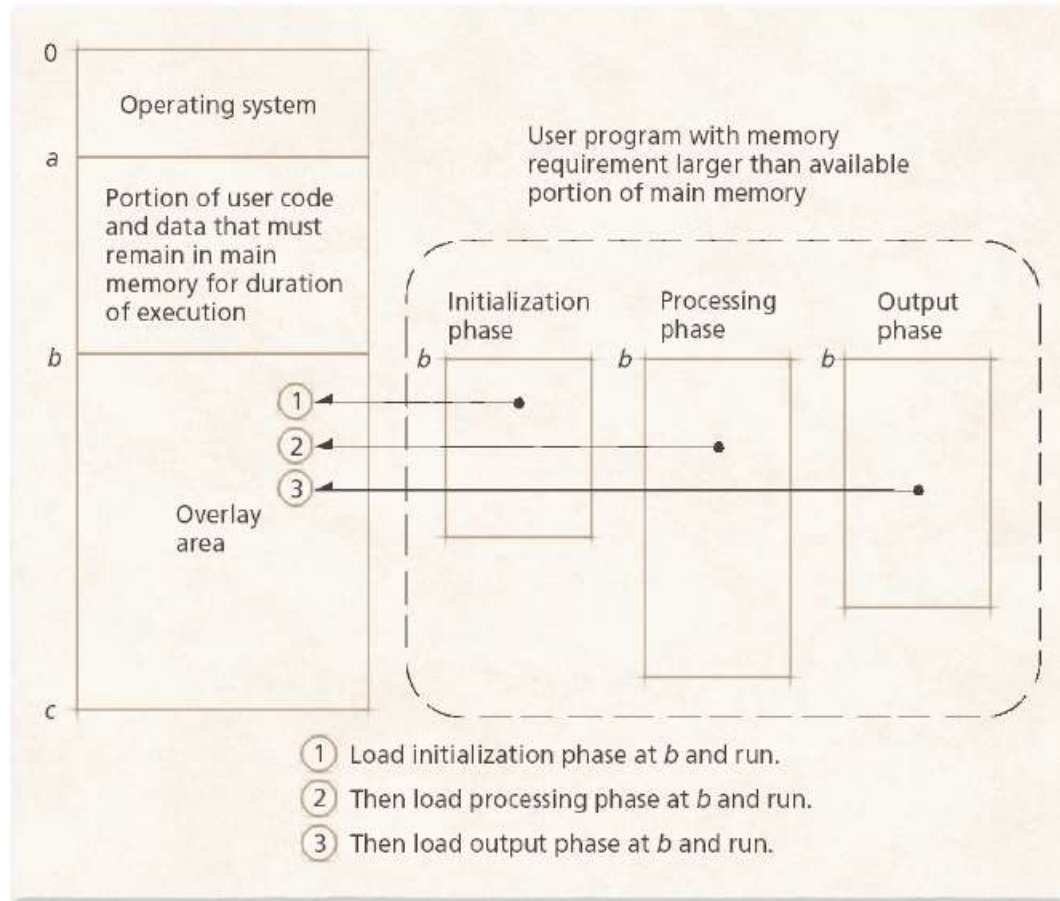
Μονοπρογραμματισμός



- Τρεις απλοί τρόποι οργάνωσης μνήμης σε ΛΣ με μια διεργασία χρήστη
- Η προσασία μνήμης δεν τίθεται ως πρόβλημα



Επικαλύψεις (overlays)



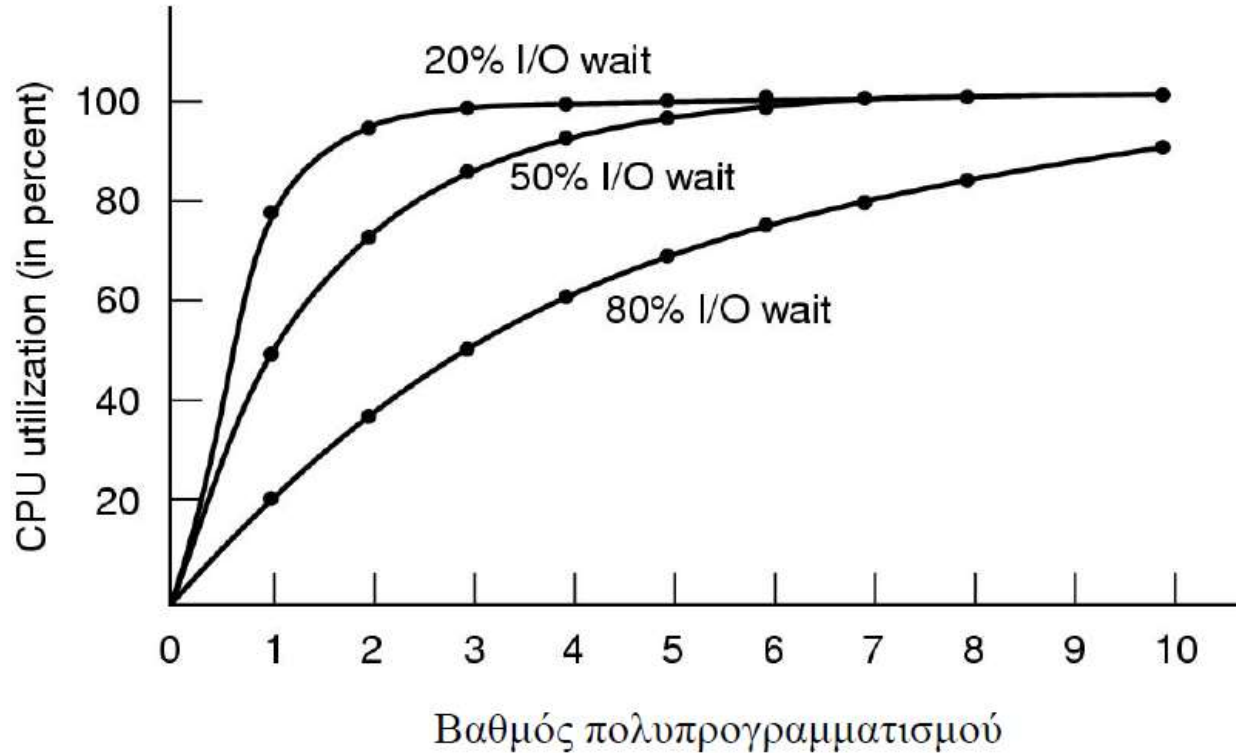


3. Μνήμη και πολυπρογραμματισμός

- Η χρήση του επεξεργαστή από μία διεργασία διακόπτεται συχνά, λόγω της ανάγκης για λειτουργίες I/O που είναι υπερβολικά αργές συγκρινόμενες με την ταχύτητα της 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 I/O κάποια άλλη να χρησιμοποιεί την CPU ώστε να αυξάνεται το ποσοστό χρήσης της CPU και η απόδοση (throughput) του συστήματος.



Χρήση CPU – Πλήθος διεργασιών



$CPU = 1 - p^v$ p : πιθανότητα I/O, v : πλήθος διεργασιών



Πλήθος προγραμμάτων στη μνήμη

- 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
 - Περισσότερη αναμονή για I/O σημαίνει μικρότερη χρήση επεξεργαστή

Στο υπόλοιπο της διάλεξης θεωρούμε ότι σε κάθε διεργασία εκχωρείται συνεχής χώρος στη μνήμη



4 Τμηματοποίηση σταθερού μεγέθους

- Η μνήμη χωρίζεται σε τμήματα, τα οποία παραμένουν σταθερά σε αριθμό και μέγεθος κατά τη λειτουργία του συστήματος
- Κάθε διεργασία καταλαμβάνει ένα μόνο τμήμα
- Το μέγιστο πλήθος των διεργασιών στη μνήμη είναι ίσο με το πλήθος των τμημάτων
- Δύο προσεγγίσεις
 - Ίσα τμήματα: Όλα τα τμήματα έχουν ίσο μέγεθος
 - Άνισα τμήματα: Τα τμήματα έχουν διαφορετικά μεγέθη



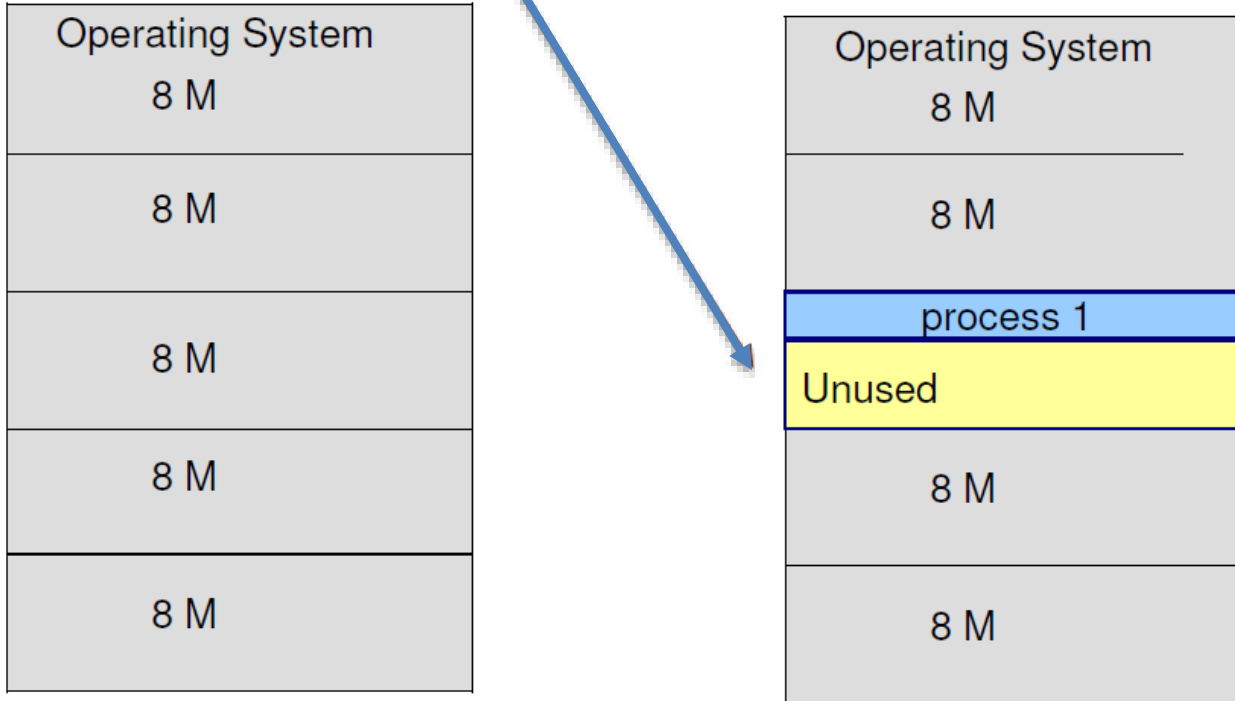
4.1 Τμηματοποίηση σταθερού μεγέθους – Ίσα Τμήματα

- Τμηματοποίηση σταθερού μεγέθους (fixed Partitioning) με ίσα τμήματα
 - 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 Λ.Σ. 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. Κάθε πρόγραμμα, όσο μικρό και να είναι, καταλαμβάνει ένα ολόκληρο τμήμα.
- Ο ανεκμετάλλετος χώρος εσωτερικά σε ένα τμήμα αναφέρεται ως εσωτερικός κατακερματισμός (internal fragmentation).



Ίσα τμήματα

- Εσωτερικός κατακερματισμός – μέρος του τμήματος που δεν χρησιμοποιείται





Ίσα τμήματα

- Πλεονεκτήματα
 - Μικρή επιβάρυνση στο Λ.Σ.
- Μειονεκτήματα
 - Ανεπαρκής χρήση της μνήμης λόγω του εσωτερικού κατακερματισμού
 - Οι μικρές διεργασίες δεν χρησιμοποιούν αποτελεσματικά τον χώρο των τμημάτων (εσωτερικός κατακερματισμός)



4.2 Τμηματοποίηση σταθερού μεγέθους – Άνισα Τμήματα

- Τμηματοποίηση σταθερού μεγέθους με άνισα τμήματα
 - Μειώνει τα προβλήματα της τμηματοποίησης ίσων τμημάτων
 - Κάθε διεργασία με μέγεθος μικρότερο ή ίσο με το μέγεθος του τμήματος μπορεί να φορτωθεί στο διαθέσιμο τμήμα (προτιμάται το τμήμα που σπαταλά το λιγότερο χώρο)
 - Αν όλα τα τμήματα είναι γεμάτα, το Λ.Σ. μπορεί να κάνει εναλλαγή μιας διεργασίας
 - Ένα πρόγραμμα είναι πιθανό να μη χωρά σε κανένα τμήμα. Ο προγραμματιστής πρέπει να σχεδιάσει το πρόγραμμα με επικαλύψεις
- Πλεονεκτήματα
 - Μειώνεται ο εσωτερικός κατακερματισμός.
 - Αποτελεσματικότερη χρήση της κύριας μνήμης, σε σχέση με τη μέθοδο των ίσων τμημάτων



Κατακερματισμός

- Εσωτερικός κατακερματισμός
 - Είναι η μνήμη που δεν χρησιμοποιείται (δαπανάται) και είναι ορατή μόνον 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 κατακερματισμός
 - Είναι η μνήμη που δεν 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.

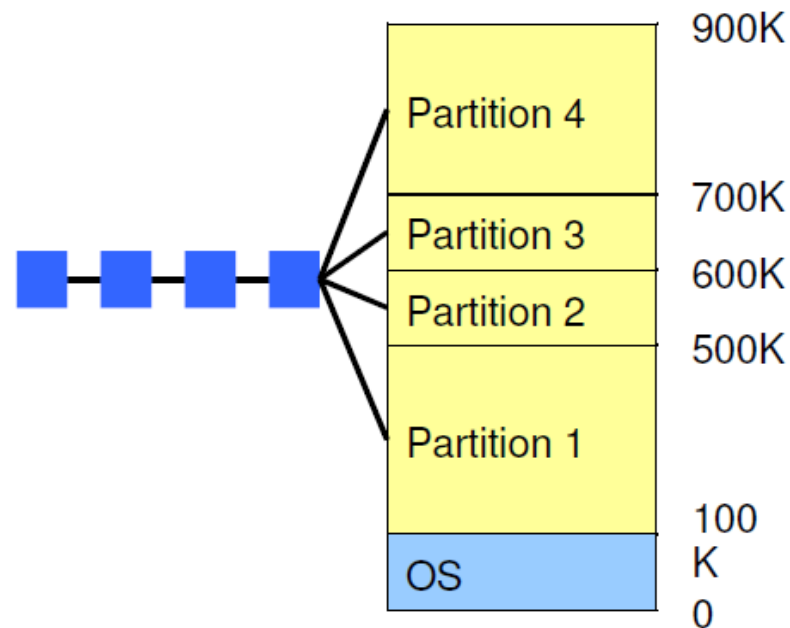
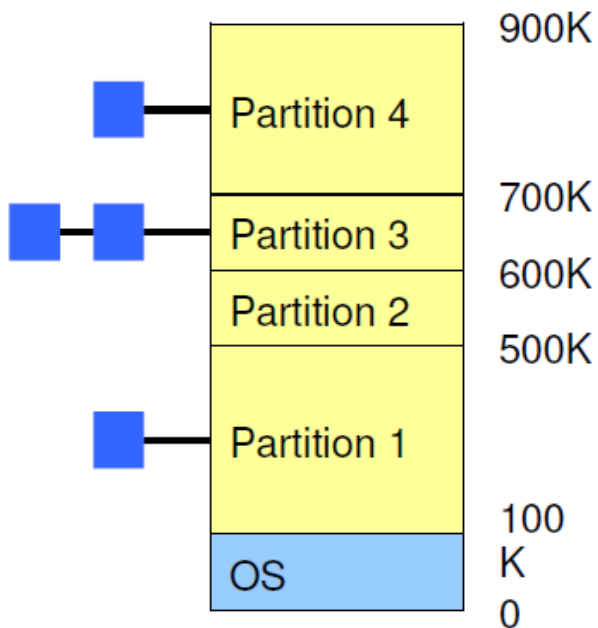


Αλγόριθμοι τοποθέτησης

- Ίσου μεγέθους τμήματα
 - Επειδή όλα τα τμήματα είναι ίσου μεγέθους, δεν έχει σημασία ποιο χρησιμοποιείται
 - Αν είναι όλα κατειλημμένα γίνεται εναλλαγή ή αναμονή
- Διαφορετικού μεγέθους τμήματα
 - ✓ Ουρά για κάθε τμήμα
 - Κάθε διεργασία τοποθετείται στην ουρά που αντιστοιχεί στο μικρότερο τμήμα στο οποίο χωρά.
 - Οι διεργασίες αντιστοιχούνται με τρόπο ώστε να ελαχιστοποιείται η σπατάλη μνήμης μέσα σε ένα τμήμα (εσωτερικός κατακερματισμός)
 - Μπορεί να υπάρχει ελεύθερη μνήμη και διεργασίες να αναμένουν
 - ✓ Μια μοναδική ουρά για όλες τις διεργασίες
 - Όταν η διεργασία πρέπει να φορτωθεί στη μνήμη επιλέγεται το μικρότερο διαθέσιμο τμήμα που τη χωρά
 - Καλύτερη ικανότητα για τη βελτιστοποίηση χρήσης της CPU



Αλγόριθμοι τοποθέτησης – Άνισα τμήματα



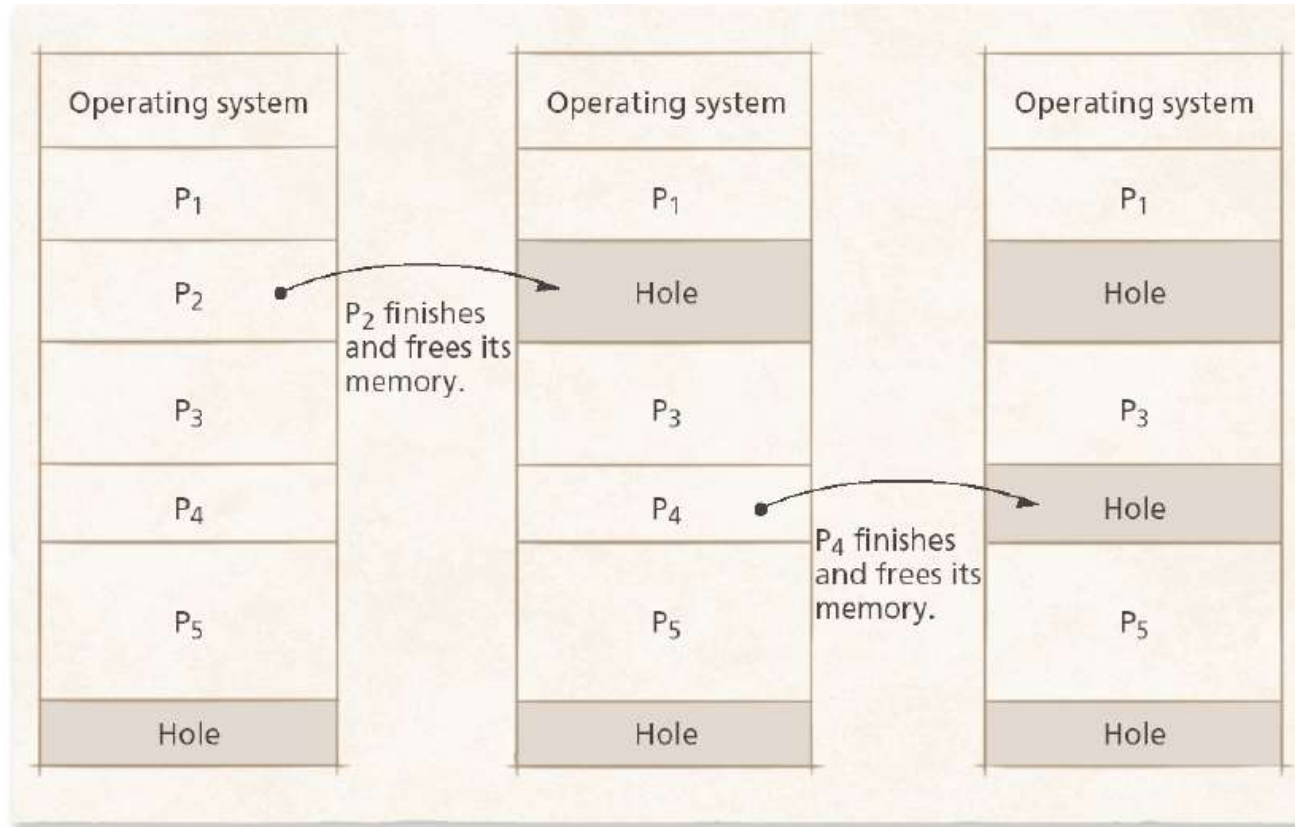


5. Δυναμική τμηματοποίηση

- 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





Αλγόριθμοι τοποθέτησης

- Το ΛΣ 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
 - Επειδή βρίσκει το μικρότερο block για τη διεργασία, η κύρια μνήμη γεμίζει γρήγορα από blocks που είναι πολύ μικρά και η συμπίεση πραγματοποιείται πιο συχνά

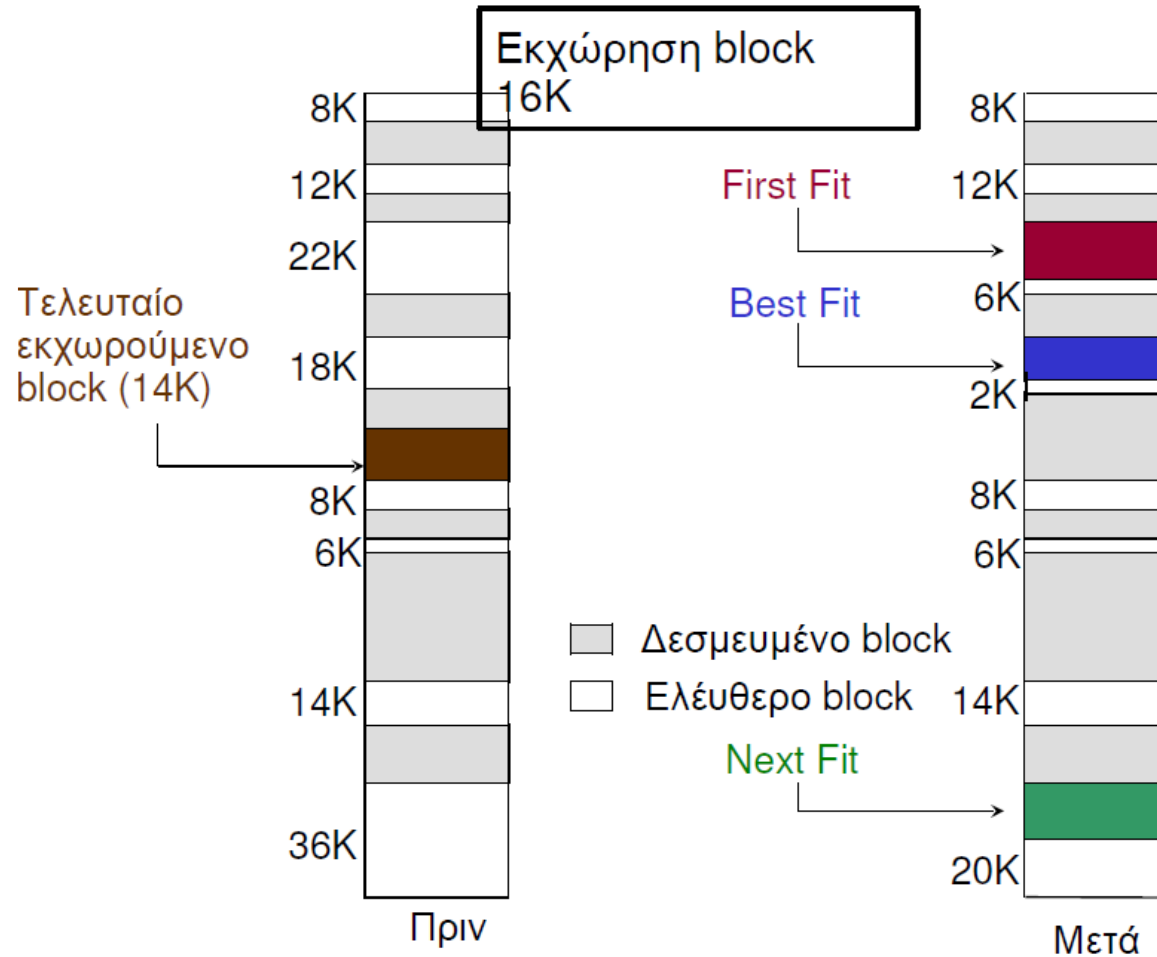


Αλγόριθμοι τοποθέτησης

- 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
- 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 συμπύεση απαιτείται για να αποκτηθεί ένα μεγάλο block στο τέλος της μνήμης.
 - Παρόμοια επίδοση με τον αλγόριθμο πρώτης τοποθέτησης

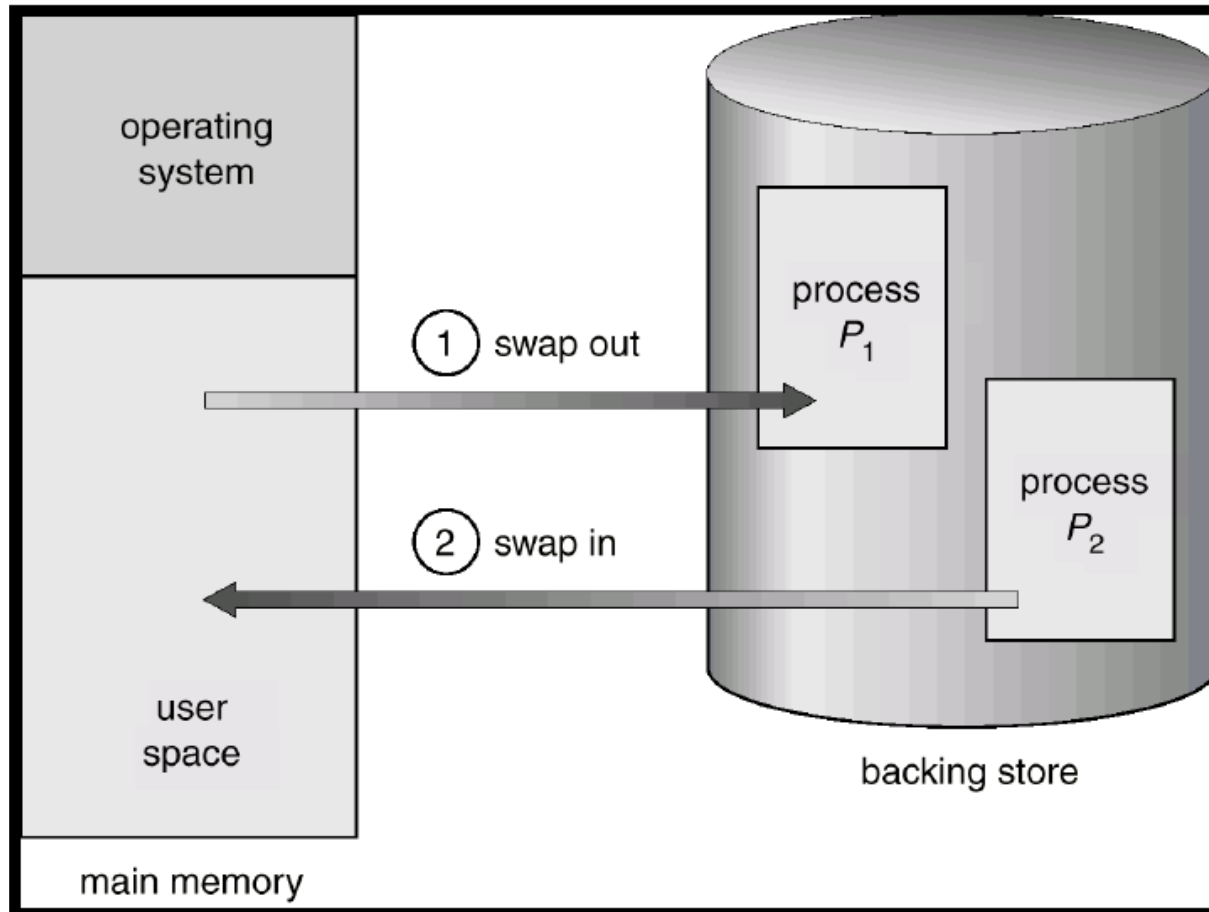


Παράδειγμα





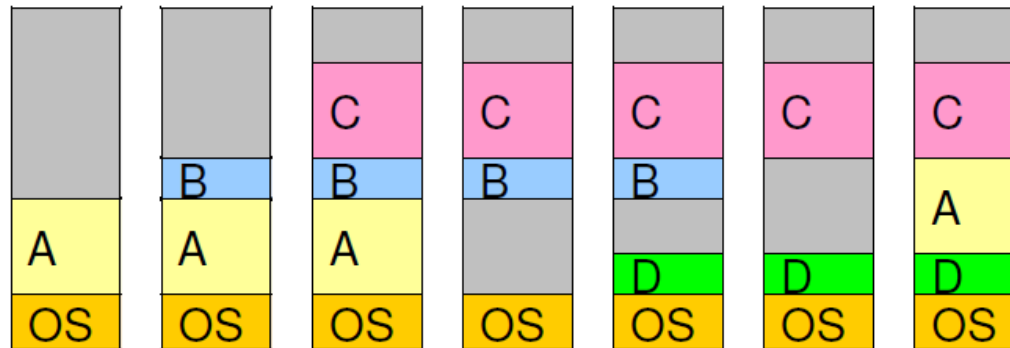
6. Εναλλαγή (swapping)





Εναλλαγή (Swapping)

- Η εκχώρηση μνήμης αλλάζει καθώς
 - Έρχονται νέες διεργασίες στη μνήμη
 - Διεργασίες εγκαταλείπουν τη μνήμη
 - Εναλλάσσονται στο δίσκο
 - Ολοκληρώνουν την εκτέλεσή τους
- Πραγματοποιείται συνεχής εκχώρηση μνήμης στις διεργασίες
- Συνήθως εφαρμόζεται σε συνδυασμό με τη δυναμική τμηματοποίηση
- Οι γκρι περιοχές είναι μνήμη που δεν χρησιμοποιείται





Περιορισμοί της εναλλαγής

- Προβλήματα με την εναλλαγή
 - Η διεργασία πρέπει να χωρά στη φυσική μνήμη (αδύνατη η εκτέλεση μεγαλύτερων διεργασιών) λόγω της συνεχούς εκχώρησης
 - Η μνήμη κατακερματίζεται και απαιτείται συμπίεση για την επανασυναρμολόγηση μεγαλύτερων ελεύθερων περιοχών
 - Οι διεργασίες μπορούν να βρίσκονται και στη μνήμη και στο δίσκο
- Οι επικαλύψεις επιλύουν το πρώτο πρόβλημα
 - Κατατέμνουν τη διεργασία στη διάρκεια του χρόνου (κυρίως τα δεδομένα)
 - Δεν επιλύουν το πρόβλημα του κατακερματισμού



Άσκηση 1

- 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 αχρησιμοποίητα τμήματα μνήμης, οι λευκές τα κενά 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.



Άσκηση 2

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



Βιβλιογραφία

Λειτουργικά Συστήματα, 8η Έκδοση, Stallings William

Λειτουργικά Συστήματα 9η Εκδ., Abraham Silberschatz, Peter Baer Galvin, Greg Gagne



Σημείωμα Αναφοράς

Copyright Τεχνολογικό Ίδρυμα Ηπείρου. Δημήτριος Λιαροκάπης.
Λειτουργικά Συστήματα.

Έκδοση: 1.0 Άρτα, 2015. Διαθέσιμο από τη δικτυακή διεύθυνση:
<http://eclass.teiep.gr/courses/COMP116/>



Σημείωμα Αδειοδότησης

Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά Δημιουργού-Μη Εμπορική Χρήση-Όχι Παράγωγα Έργα 4.0 Διεθνές [1] ή μεταγενέστερη. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, Διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό και τα οποία αναφέρονται μαζί με τους όρους χρήσης τους στο «Σημείωμα Χρήσης Έργων Τρίτων».



Ο δικαιούχος μπορεί να παρέχει στον αδειοδόχο ξεχωριστή άδεια να χρησιμοποιεί το έργο για εμπορική χρήση, εφόσον αυτό του ζητηθεί.

[1] <http://creativecommons.org/licenses/by-nc-nd/4.0/deed.el>



Τέλος Ενότητας

Επεξεργασία: Ευάγγελος Καρβούνης
Άρτα, 2015



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ



Τέλος Ενότητας

Διαχείριση Μνήμης



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

