

Ιόνιο Πανεπιστήμιο – Τμήμα Πληροφορικής  
Εισαγωγή στην Επιστήμη των Υπολογιστών  
2016-17

# Λειτουργικά Συστήματα

(διαχείριση επεξεργαστή, μνήμης και Ε/Ε)

<http://mixstef.github.io/courses/csintro/>

Μ.Στεφανιδάκης



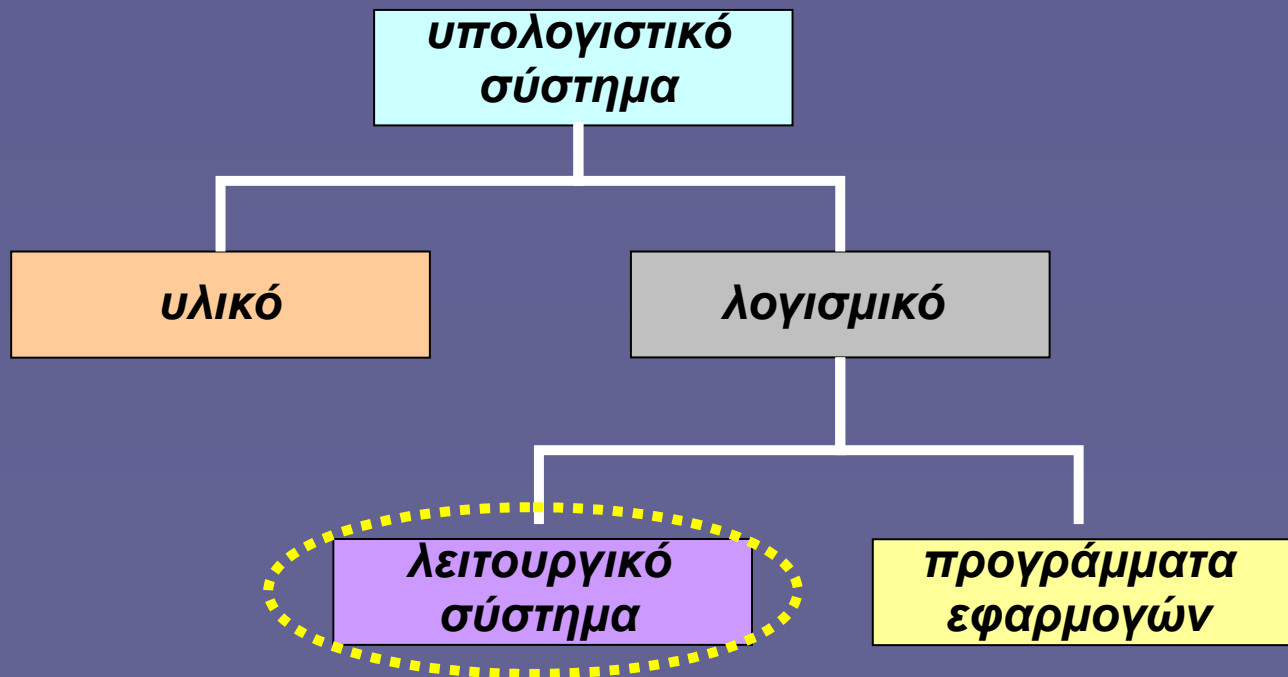
# Τι είναι ένα λειτουργικό σύστημα (ΛΣ);

- Εισαγωγή

- Operating System (OS)
- Λογισμικό – ο “γενικός επιβλέπων” ενός υπολογιστικού συστήματος
  - Εκτέλεση προγραμμάτων εφαρμογών
  - Διαχείριση υπολογιστικών πόρων
    - Επεξεργαστής
    - Μνήμη
    - Συσκευές E/E
  - Ενδιάμεσο μεταξύ χρήστη και υπολογιστή
  - Παρέχει κοινές/συχνά χρησιμοποιούμενες λειτουργίες εφαρμογών
    - “Βιβλιοθήκες” (libraries) λειτουργικού συστήματος

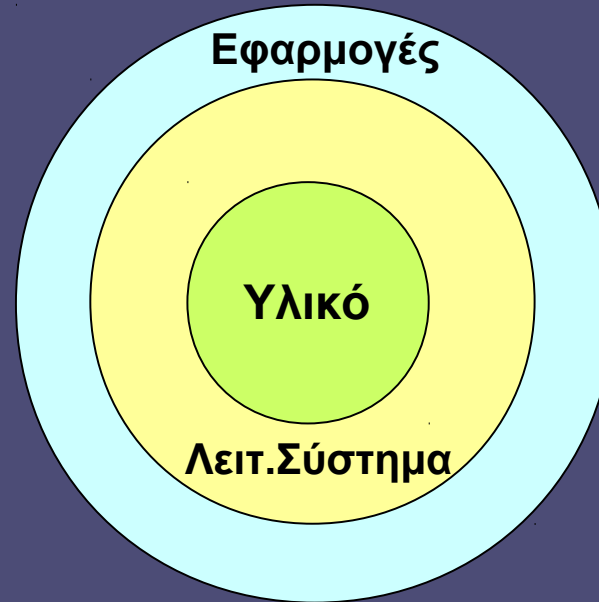
# Το Λειτουργικό Σύστημα ως μέρος του υπολογιστή

- Εισαγωγή



# Μια άλλη απεικόνιση

- Εισαγωγή



- Το Λειτουργικό Σύστημα ως κέλυφος (shell) μεταξύ των εφαρμογών χρήστη και του υλικού
- Οι εφαρμογές χρήστη δεν έχουν δικαιώματα διαχείρισης του υλικού

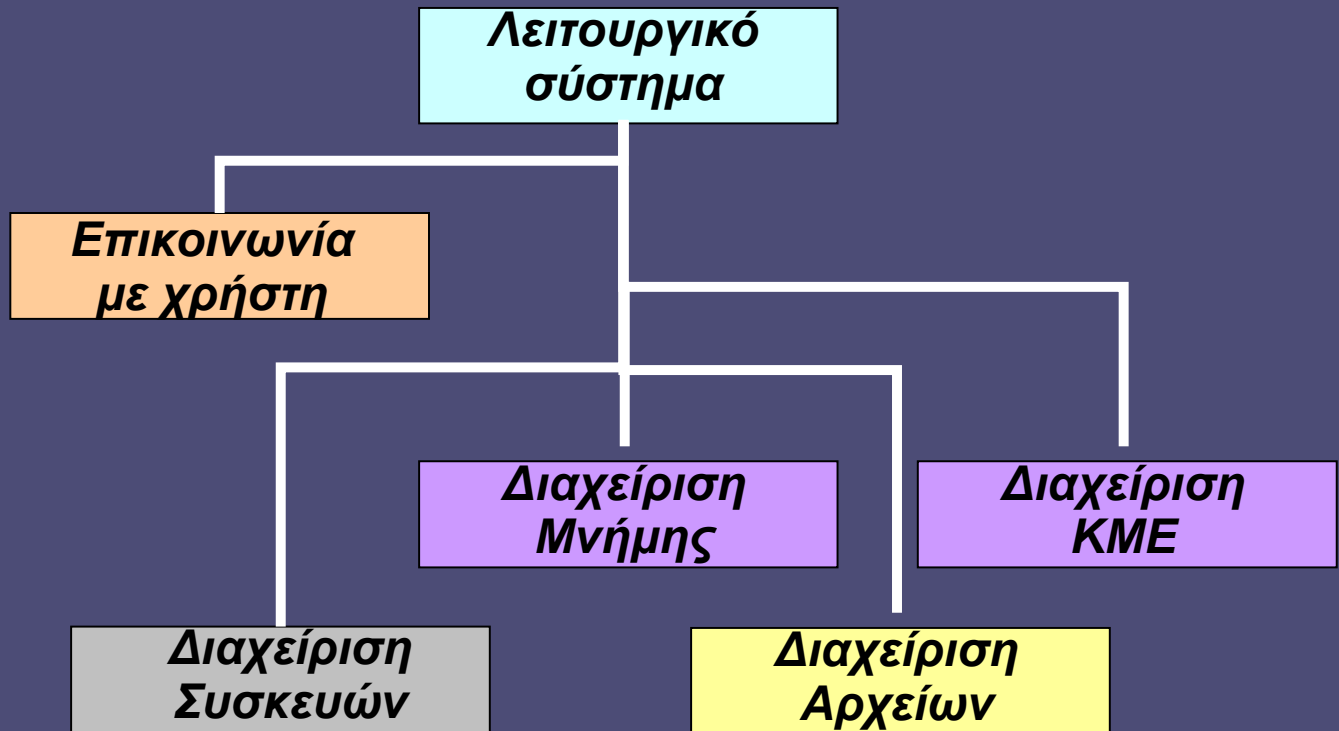
# Η εξέλιξη των Λειτουργικών Συστημάτων

- Εισαγωγή

- **Ανάλογα με την εξέλιξη των υπολογιστών:**
- **Τα πρώτα μεγάλα συστήματα (mainframes)**
  - Λ.Σ. δέσμης (batch operating systems)
- **Συστήματα με πολυπρογραμματισμό (multiprogramming)**
  - Λ.Σ. με δυνατότητες χρονοπρογραμματισμού
- **Προσωπικοί Υπολογιστές**
  - Λ.Σ. ενός χρήστη (single user operating system)
- **Παράλληλα συστήματα (πολλές ΚΜΕ)**
  - Σύνθετα Λ.Σ. για κατανομή εργασίας
- **Κατανεμημένα συστήματα (μέσω δικτύου-internet)**
  - Λ.Σ. με πρόσθετες ικανότητες (συντονισμός, μηνύματα, ασφάλεια δεδομένων..)

# Διαχείριση πόρων

- Εισαγωγή



- Είναι η επικοινωνία με τον χρήστη μέρος του Λ.Σ;

# Μονοπρογραμματισμός

- Εισαγωγή
- Διαχείριση μνήμης

- Τα πρώτα χρόνια των υπολογιστών...
- Στην μνήμη βρίσκονται
  - Ένα μοναδικό πρόγραμμα
  - Το λειτουργικό σύστημα
- Απλή διαχείριση μνήμης
- Αν το μέγεθος του προγράμματος είναι πολύ μεγάλο;
- Πότε μπορεί να εκτελεστεί άλλο πρόγραμμα;

**Λειτουργικό  
Σύστημα**

**Πρόγραμμα:  
κώδικας και  
δεδομένα**

**μνήμη**

# Πολυπρογραμματισμός

- Εισαγωγή
- Διαχείριση μνήμης

- Πολλά προγράμματα στη μνήμη
- “Ταυτόχρονη” εκτέλεση
  - Στην πραγματικότητα διαδοχική εκτέλεση εντολών από κάθε πρόγραμμα
  - Φαινομενικά τα προγράμματα εκτελούνται “παράλληλα”
- Οι υπολογιστικοί πόροι πρέπει να μοιράζονται μεταξύ των προγραμμάτων
  - Αναγκαία η διαιτησία
  - Ρόλος του Λειτουργικού Συστήματος!

**Λειτουργικό  
Σύστημα**

**Πρόγραμμα1**

**Πρόγραμμα2**

**Πρόγραμμα3**

**Πρόγραμμα4**

**μνήμη**



# Πολυπρογραμματισμός (συνέχεια)

- Εισαγωγή
- Διαχείριση μνήμης

- Το Λειτουργικό Σύστημα διατηρεί πληροφορία για κάθε εκτελούμενο πρόγραμμα (διεργασία)
  - Process Control Block (PCB) ή Task Control Block (TCB)
- Όταν διακόπτεται η εκτέλεση:
  - Αποθήκευση τιμής program counter (PC)
    - Τρέχουσα διεύθυνση εκτελούμενης εντολής
  - Αποθήκευση τιμών καταχωρητών
- Όταν ξεκινά πάλι η εκτέλεση:
  - Αποκατάσταση αποθηκευμένων τιμών
    - Το πρόγραμμα συνεχίζει την εκτέλεση από το σημείο που διακόπηκε

# Τεχνικές διαχείρισης μνήμης

- Εισαγωγή
- Διαχείριση μνήμης

- Πώς θα εξασφαλιστεί η διαθεσιμότητα της μνήμης για τα προγράμματα που εκτελούνται “ταυτόχρονα”
- Που βρίσκεται το πρόγραμμα;
  - Χωρίς εναλλαγή
    - Το πρόγραμμα παραμένει συνεχώς στη μνήμη
  - Με εναλλαγή (swapping)
    - Μέρος του προγράμματος μπορεί να εναλλάσσει θέση μεταξύ μνήμης και δίσκου κατά την εκτέλεση

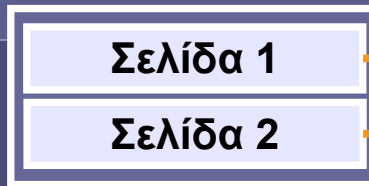
# Χωρίς εναλλαγή: διαμέριση (partitioning)

- Εισαγωγή
- Διαχείριση μνήμης

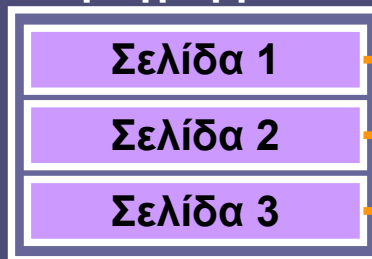
- Κάθε πρόγραμμα έχει τον δικό του χώρο (διαμέριση) στη μνήμη
  - Όλο το πρόγραμμα σε συνεχόμενες θέσεις μνήμης
  - Αντικατάσταση προγραμμάτων μετά τον τερματισμό
  - Εισαγωγή νέων
  - Πιθανή δημιουργία κενών τμημάτων
  - Αντιμετώπιση κενών με συμπίκνωση (compaction)
    - Αργή διαδικασία!
  - Το Λ.Σ. διατηρεί λίστα χρησιμοποιούμενων τμημάτων μνήμης

# Με εναλλαγή: Σελιδοποίηση (1)

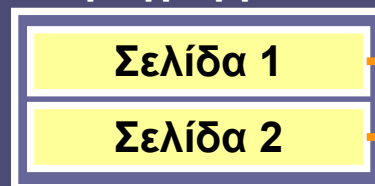
Πρόγραμμα 1



Πρόγραμμα 2



Πρόγραμμα 3



Κύρια μνήμη



- Εισαγωγή
- Διαχείριση μνήμης

- Κάθε πρόγραμμα χωρίζεται σε σελίδες (**pages**)
- Η μνήμη χωρίζεται σε πλαίσια (**frames**)
  - Σελίδες και πλαίσια: ίδιο μέγεθος
  - Αποδοτικότερη χρήση μνήμης (λιγότερα κενά)
  - Πώς λύνεται το πρόβλημα της ανεπαρκούς μνήμης;

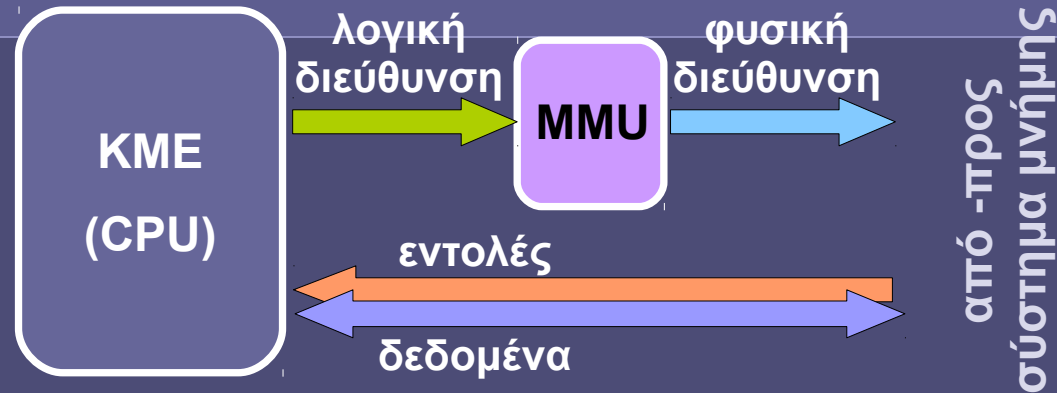
# Σελιδοποίηση (2)

- Εισαγωγή
- Διαχείριση μνήμης

- Οι σελίδες έχουν προκαθορισμένο μέγεθος
  - π.χ. 4KBytes
- Μπορούν να τοποθετηθούν σε οποιοδήποτε πλαίσιο μνήμης
- Ενδεχομένως να μην βρίσκονται όλες στη μνήμη
  - Ανάκληση από δίσκο όταν τις χρειαστεί το πρόγραμμα
  - Σελιδοποίηση κατ'απαίτηση (demand paging)
- Πρακτικά...
  - Το πρόγραμμα βλέπει διαφορετική διεύθυνση μνήμης από την πραγματική (φυσική) του πλαισίου
  - Εικονική μνήμη (virtual memory)
  - Απαιτείται η τήρηση πινάκων αντιστοίχισης διευθύνσεων
  - Για τη λειτουργία της εικονικής μνήμης είναι απαραίτητη η συνδυασμένη υποστήριξη από το υλικό (KME) και το λειτουργικό σύστημα

# Εικονική μνήμη: τι παρέχει το υλικό

- Εισαγωγή
- Διαχείριση μνήμης



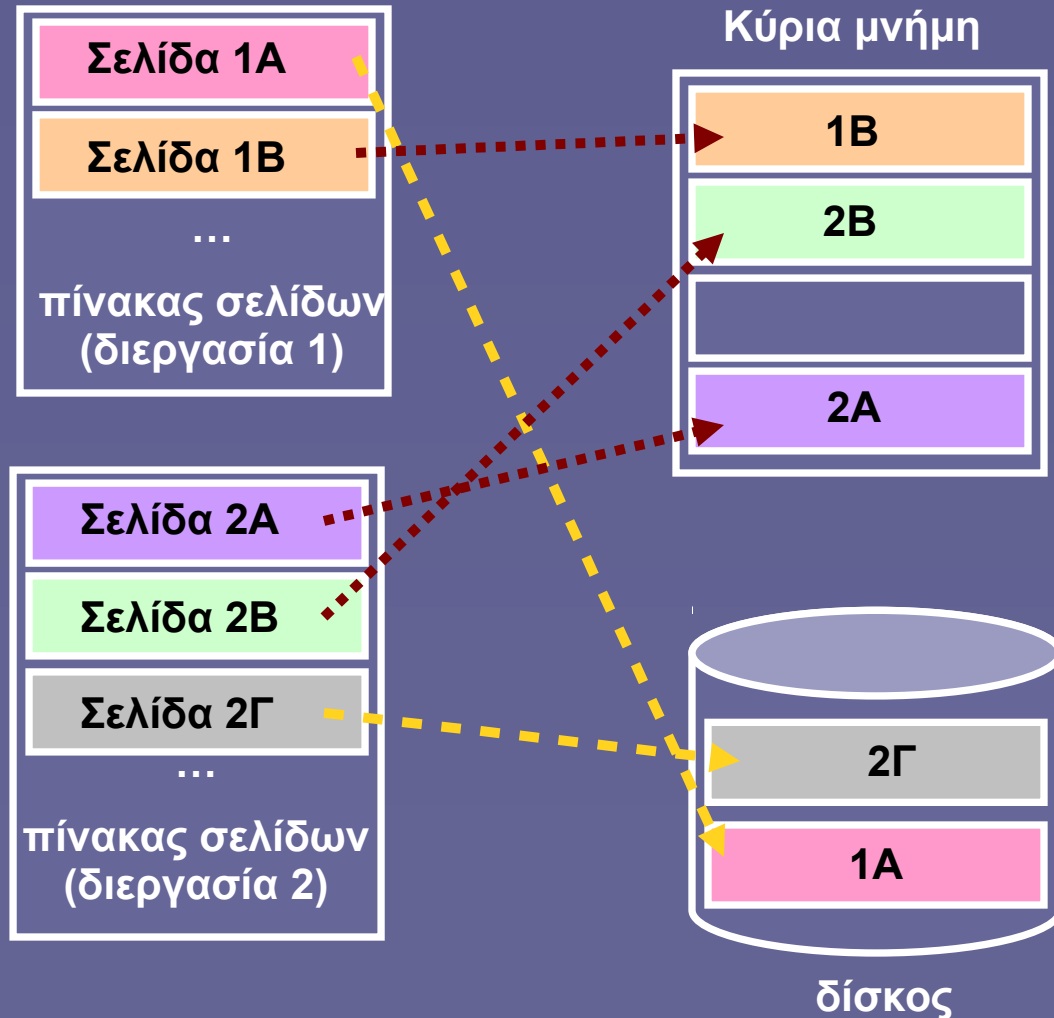
- Ένα πρόγραμμα “βλέπει” **λογικές** διευθύνσεις
  - Εικονική Μνήμη (virtual memory)
  - Μετάφραση σε **φυσικές** διευθύνσεις μνήμης
  - Από το σύστημα διαχείρισης μνήμης (**memory management unit – MMU**) που βρίσκεται επίσης μέσα στον επεξεργαστή
    - Για τη μετάφραση, αναζήτηση σε **πίνακες σελίδων**
    - Αποθήκευση των πιο πρόσφατων μεταφράσεων στην KME

# Εικονική μνήμη: τι παρέχει το Λ.Σ.

- Εισαγωγή
- Διαχείριση μνήμης

Κάθε διεργασία έχει τους δικούς του πίνακες σελίδων!

Κατά την εναλλαγή διεργασιών αλλάζει και ποιος πίνακας θα χρησιμοποιηθεί για τη μετάφραση



# Σελιδοποίηση κατ'απαίτηση

- Εισαγωγή
- Διαχείριση μνήμης

Σε πολλά συστήματα η μετάφραση των διευθύνσεων γίνεται σε 2 επίπεδα:  
Ο χώρος προγράμματος χωρίζεται σε τμήματα (segments) και τα τμήματα με τη σειρά τους χωρίζονται σε σελίδες

- Οι σελίδες των προγραμμάτων (κώδικας-δεδομένα) βρίσκονται αρχικά μόνο στον δίσκο
- Το ΛΣ τις σημειώνει ως “**απούσες**” από τη μνήμη
- Όταν προσπελαστεί μια “απούσα” σελίδα, δημιουργείται ένα σφάλμα εκτέλεσης (**page fault**)...
- ...και το ΛΣ τη φορτώνει σε ένα πλαίσιο στη μνήμη
- Ενδεχομένως εκτοπίζοντας πίσω στον δίσκο μια άλλη σελίδα από τη μνήμη
  - Η τελευταία σημειώνεται ως “**απούσα**”
- **Page faults**: μεγάλο κόστος σε κύκλους αναμονής
  - 1-10M κύκλοι



# Προγράμματα, εργασίες και διεργασίες

- Εισαγωγή
- Διαχείριση μνήμης
- Διεργασίες

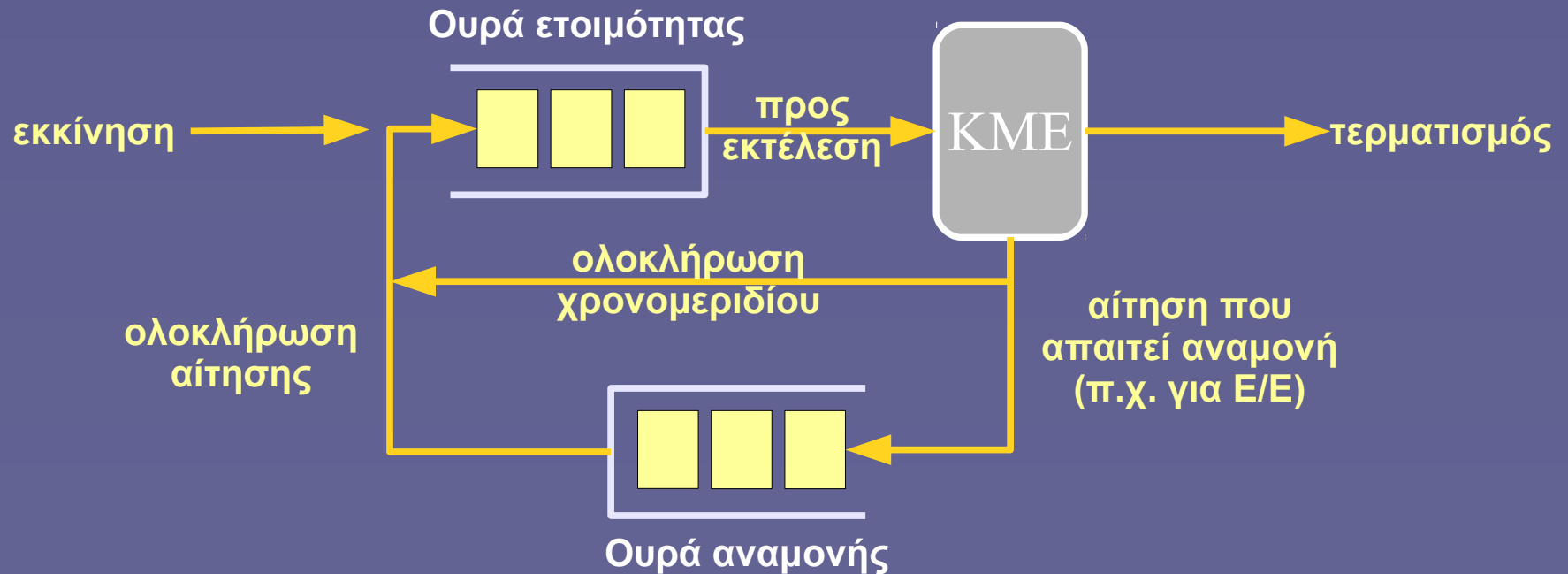
- **Πρόγραμμα (program)**
  - Ανενεργό σύνολο εντολών (στον δίσκο)
- **Εργασία (job)**
  - Πρόγραμμα που έχει επιλεγεί για εκτέλεση
  - Στον δίσκο ή στη μνήμη
- **Διεργασία (process)**
  - Πρόγραμμα υπό εκτέλεση
  - Στη μνήμη
  - Εκτελείται ή αναμένει για χρήση πόρων
  - Μπορεί να διακοπεί οποιαδήποτε στιγμή “παρά τη θέλησή της” από το Λ.Σ. (προεκτοπισμός – preemption)

# Χρονοδρομολόγηση (scheduling)

- Εισαγωγή
- Διαχείριση μνήμης
- Διεργασίες

- **Καταστάσεις διεργασίας**
  - Έτοιμη προς εκτέλεση (ready)
  - Εκτελούμενη (running)
  - Σε αναμονή (waiting/blocked)
    - Για τη χρήση υπολογιστικών πόρων
- **Όταν πολλές διεργασίες είναι έτοιμες, ποια θα εκτελεστεί;**
  - Απόφαση χρονοδρομολογητή (scheduler)
    - είναι μέρος του Λειτουργικού Συστήματος
  - Κριτήριο Επιλογής
    - “Όποιος ήρθε πρώτος” (first-in first-out – FIFO)
    - Χρησιμοποιώντας προτεραιότητες
    - Ανάλογα με προθεσμίες (συστήματα real-time)

# Ουρές διαχείρισης διεργασιών

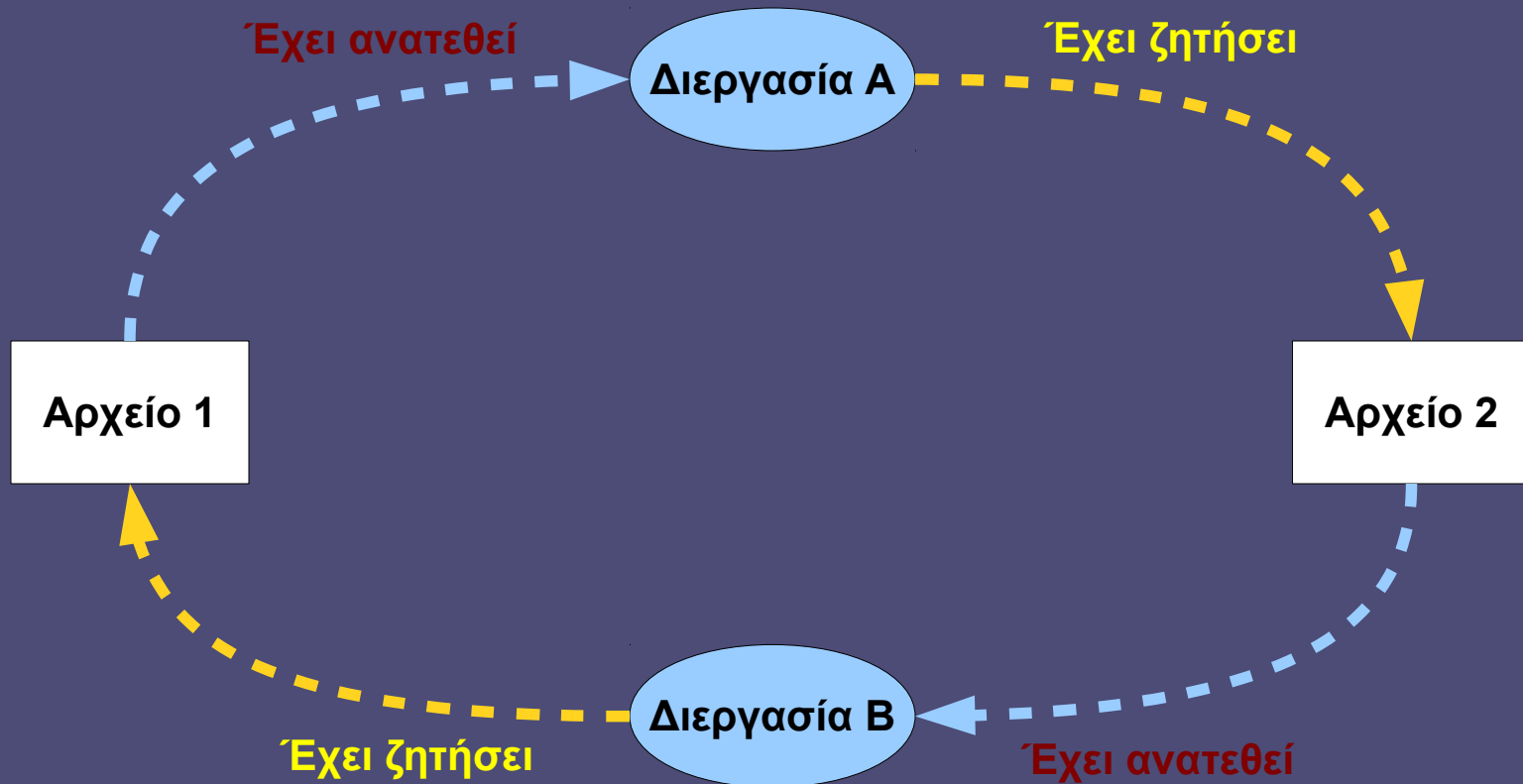


# Συγχρονισμός διεργασιών

- Εισαγωγή
- Διαχείριση μνήμης
- Διεργασίες

- Στη χρήση διαμοιραζόμενων πόρων
  - Οι κοινοί πόροι μπορούν να χρησιμοποιούνται μόνο από μία διεργασία κάθε στιγμή
- Αναμονή υπολοίπων διεργασιών
- Εμφάνιση καταστάσεων σύγκρουσης
  - Αδιέξοδο
  - Λιμοκτονία

# Αδιέξοδο



- Τα αρχεία είναι κοινόχρηστοι πόροι!

# Αδιέξοδο (συνέχεια)

- Εισαγωγή
- Διαχείριση μνήμης
- Διεργασίες

- **Συνθήκες εμφάνισης**
  - Αμοιβαίος αποκλεισμός στη χρήση των πόρων
  - Παρακράτηση περισσότερων από έναν πόρο
  - Κυκλική αναμονή
- **Αν επιτρέψουμε την εκτέλεση διεργασίας μόνο εάν όλοι οι πόροι που ζητά είναι ελεύθεροι;**
  - Κίνδυνος λιμοκτονίας
    - Η διεργασία μπορεί να μην εκτελεστεί “ποτέ”

# Διαχείριση συσκευών

- Εισαγωγή
- Διαχείριση μνήμης
- Διεργασίες
- Διαχείριση συσκευών

- **Συσκευές εισόδου-εξόδου (E/E)**
  - Διαχείριση από ΛΣ μόνο
    - Η απευθείας προσπέλαση των συσκευών E/E είναι απαγορευμένη στις εφαρμογές χρήστη!
  - Ρύθμιση χρήσης από διεργασίες
    - Οι συσκευές E/E είναι ένα τυπικό παράδειγμα διαμοιραζόμενων (κοινόχρηστων) πόρων!
  - Παρακολούθηση ολοκλήρωσης αιτήσεων E/E
    - Και επανεκκίνηση των διεργασιών που αναμένουν τα δεδομένα E/E
  - Το εξειδικευμένο πρόγραμμα συστήματος που “συνομιλεί” με τη συσκευή E/E ονομάζεται “οδηγός” της συσκευής (device driver)