

Ιόνιο Πανεπιστήμιο – Τμήμα Πληροφορικής
Εισαγωγή στην Επιστήμη των Η/Υ
2026-27

Αναπαράσταση Μη Αριθμητικών Δεδομένων

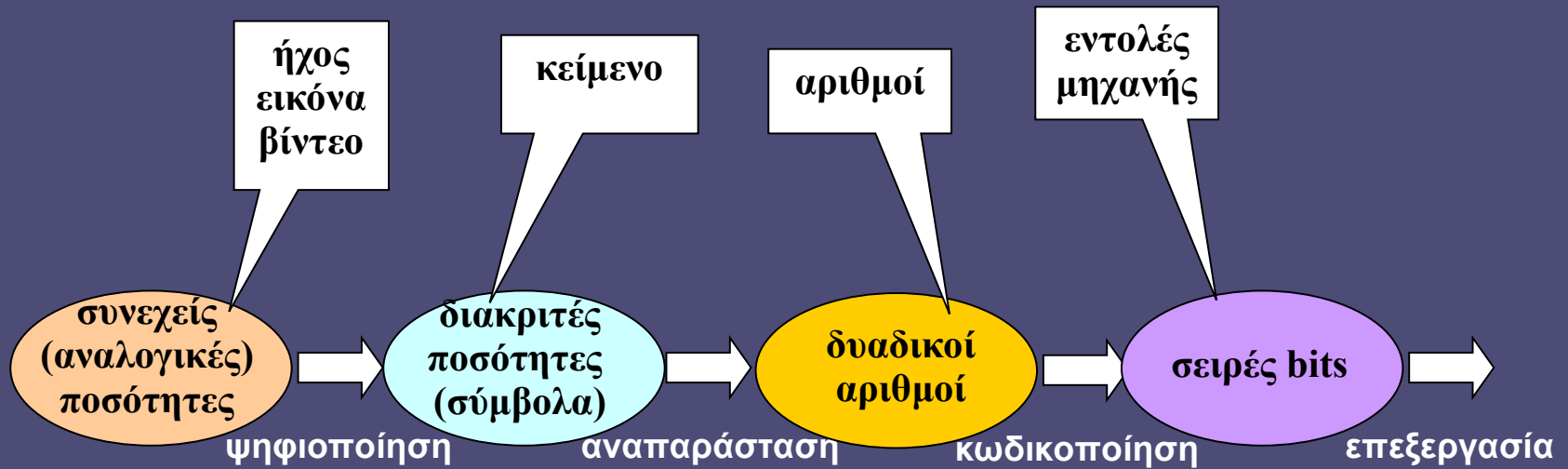
(αναπαραστάσεις κειμένου στον υπολογιστή)

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

Μ.Στεφανιδάκης



Αναπαράσταση δεδομένων



- Ψηφιοποίηση
 - Διαδικασία **μετατροπής** συνεχών τιμών σε διακριτά σύμβολα
- Αναπαράσταση
 - Διαδικασία **αντιστοίχισης** συμβόλων σε δυαδικούς αριθμούς
- Κωδικοποίηση
 - **Αποθήκευση** δυαδικών αριθμών σε σειρές bits

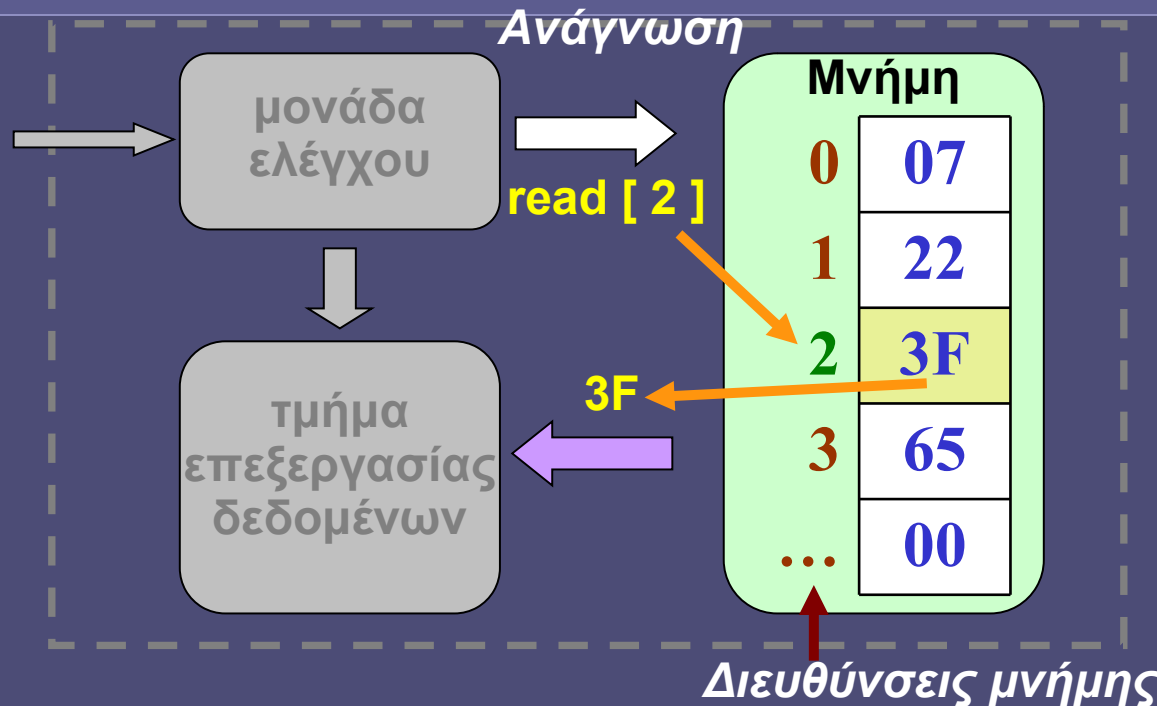
Η ερμηνεία της αναπαράστασης

- Κάπου στη μνήμη του υπολογιστή...
 - Βρίσκεται αποθηκευμένη η σειρά bits 0100110111010001
 - Πόσα σύμβολα αναπαριστά;
 - Πόσα bits ανά σύμβολο;
 - Ποιος ο τύπος των δεδομένων;
 - Ποια συγκεκριμένη ποσότητα συμβολίζει;
 - Πώς θα το χειριστεί ο υπολογιστής;
- Στα ερωτήματα αυτά μπορεί να απαντήσει μόνο ο προγραμματιστής της εφαρμογής που χειρίζεται τα δεδομένα!

Αναπαράσταση με δυαδικούς αριθμούς

- Σειρά από n bits
 - Δυαδικός αριθμός με n bits ($n \geq 1$) μπορεί να αναπαραστήσει 2^n διαφορετικά σύμβολα
- Μη αριθμητικά δεδομένα
 - Κείμενο, ήχος, εικόνα...
 - Σύνολο διαφορετικών αντικειμένων (συμβόλων)
 - Αντιστοίχιση κάθε συμβόλου σε μοναδικό δυαδικό αριθμό
 - «Αναπαράσταση»
 - Η ακριβής αντιστοίχιση ορίζεται σε ένα πρότυπο (standard)

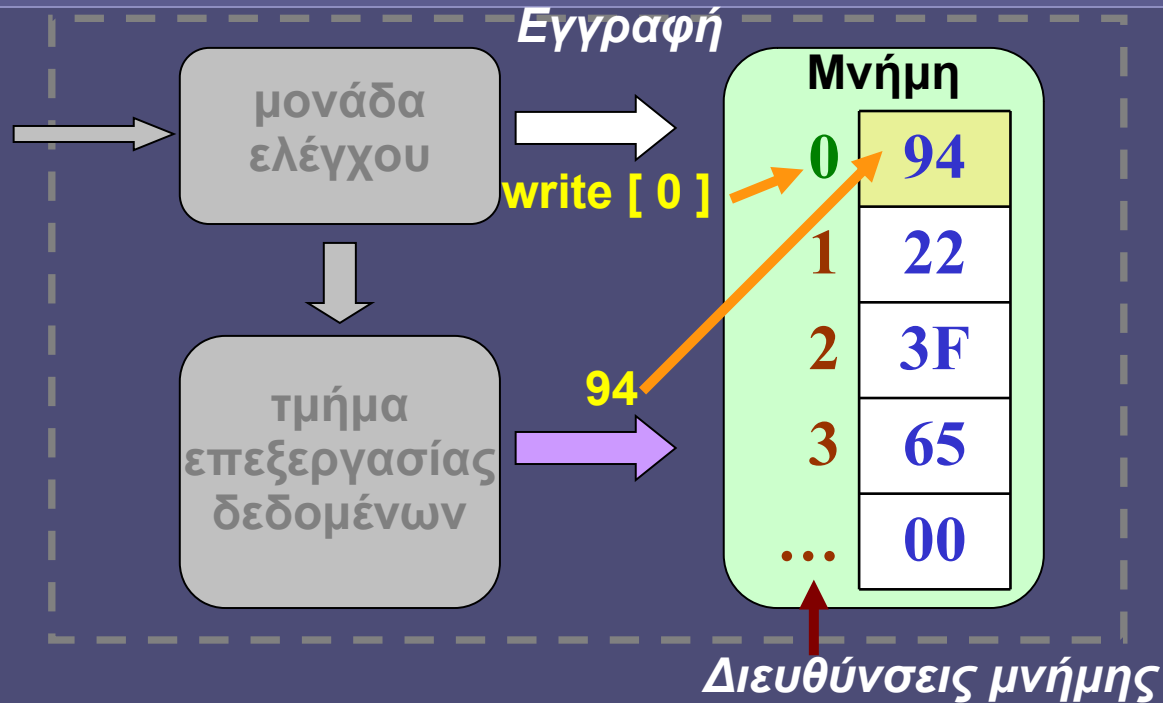
Το απλουστευμένο μοντέλο μνήμης



- Πώς «βλέπει» ένα πρόγραμμα τη μνήμη
 - Ακολουθία αποθηκευτικών θέσεων
 - Σε κάθε θέση αποθηκεύεται (συνήθως) 1 byte
 - Κάθε θέση διαθέτει **μοναδική διεύθυνση**
 - Επιλογή θέσης κατά την προσπέλαση (ανάγνωση-εγγραφή)

Με διεύθυνση των n bits, πόσες διαφορετικές θέσεις μνήμης μπορούμε να προσπελάσουμε;

Το απλουστευμένο μοντέλο μνήμης



- Στην πραγματικότητα
 - Η «μνήμη» είναι μια σύνθετη ιεραρχία **πολλών επιπέδων**
 - Οι μεταφορές δεδομένων δεν γίνονται σε μεμονωμένα bytes αλλά σε **ομάδες πολλών bytes ταυτόχρονα**

Αποθήκευση δυαδικών αριθμών στη μνήμη

- Τι συμβαίνει όταν για έναν δυαδικό αριθμό χρειάζονται **περισσότερα από ένα bytes** για να αποθηκευτούν τα ψηφία του στη μνήμη;
- Παράδειγμα: 3FC (hex) = 11 1111 1100
Απαιτούνται 2 bytes για την αποθήκευση του αριθμού αυτού

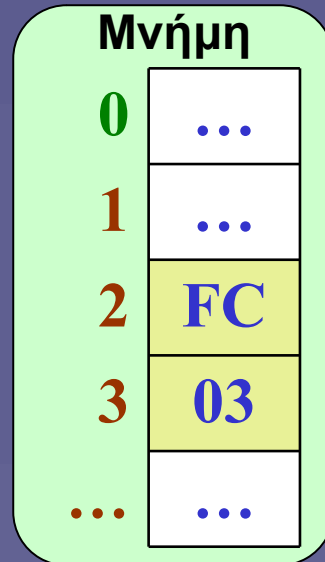
0000 0011	1111 1100
⏟	⏟
περισσότερο σημαντικό byte	λιγότερο σημαντικό byte
- Προφανώς σε συνεχόμενες θέσεις μνήμης
Αλλά: ποιο byte αποθηκεύεται πρώτο;

Αποθήκευση δυαδικών αριθμών στη μνήμη

αποθηκεύοντας το
03FC

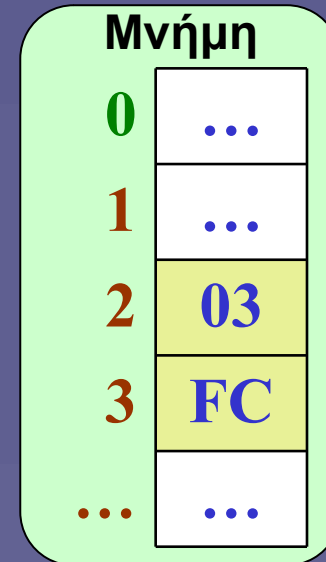
00000011 11111100

στις διευθύνσεις
μνήμης 2 και 3



“little-endian”

Το λιγότερο
σημαντικό byte στη
θέση μνήμης με
μικρότερη διεύθυνση



“big-endian”

Το περισσότερο
σημαντικό byte στη
θέση μνήμης με
μικρότερη διεύθυνση

Αρχικές αναπαραστάσεις κειμένου

- Οι πρώτες αναπαραστάσεις κειμένου
 - Στον υπολογιστή
 - 6-7 bits ανά χαρακτήρα
 - Πόσοι διαφορετικοί χαρακτήρες;
- Μη εκτυπώσιμοι χαρακτήρες
 - Χαρακτήρες ελέγχου
 - Ιδιαίτερα χρήσιμοι για τις συσκευές εξόδου της εποχής (εκτυπωτές, τηλέτυπα...)
 - Νέα γραμμή (LINE FEED – LF)
 - Επιστροφή κεφαλής εκτύπωσης (CARRIAGE RETURN – CR)
 - Καμπανάκι (BELL) κλπ

Κώδικας ASCII

(American Standard Code for Information Interchange)

- 7 bits ανά χαρακτήρα
 - 128 χαρακτήρες
 - Αναπαράσταση με τους αριθμούς 0...127
- Κανονικοί χαρακτήρες (εκτυπώσιμοι)
 - 32...47, 58...64, 91...96, 123...126 = σημεία στίξης κ.ά. (32 = SPACE)
 - 48...57 = ψηφία 0...9
 - 65...90 = κεφαλαία λατινικά (A-Z)
 - 97...122 = πεζά λατινικά (a-z)
- Χαρακτήρες ελέγχου (μη εκτυπώσιμοι)
 - 0...31, 127 – τα πιο γνωστά σε εμάς είναι: 9 (TAB), 13/10 (CR/LF, σήμανση “νέας γραμμής”)

Κείμενο σε κώδικα ASCII

- Παράδειγμα

H	a	v	e		a		n	i	c	e		d	a	y	!
72	97	118	101	32	97	32	110	105	99	101	32	100	97	121	33

- Κωδικοποίηση με 1 byte ανά χαρακτήρα
 - Δεν τίθεται θέμα “little-” ή “big-endian” αποθήκευσης γιατί κάθε χαρακτήρας είναι 1 byte

Μεταγενέστερες επεκτάσεις κώδικα ASCII

- Χρήση του ενός επιπλέον bit του byte (bit7)
 - 128 αρχικοί + 128 **νέοι** χαρακτήρες
 - 0...127 αρχικός ASCII, 128...255: **επεκταμένα αλφάβητα**
- **Επέκταση αλφαβήτων**
 - Χαρακτήρες που δεν υπάρχουν στον ASCII
 - Αρχικά: ad hoc (μη πρότυπες) λύσεις
 - Για Windows, Mac ..
 - Στη συνέχεια: διαφορετικά πρότυπα ανά γλώσσα, π.χ.:
 - ISO-8859-1: Δυτική Ευρώπη (Å, Ñ, Æ, ä, ø κλπ)
 - ISO-8859-7: Νέα Ελληνικά
 - ...και πολλά άλλα πρότυπα για τις υπόλοιπες γλώσσες
 - Δεν μπορούν να συνυπάρχουν δύο διαφορετικά πρότυπα στο ίδιο κείμενο

Κείμενο σε κώδικα ISO-8859-7

- Παράδειγμα

Γ	ε	ι	α		σ	ο	υ	!
195	229	233	225	32	243	239	245	33

- Επέκταση κώδικα ASCII

- 0...127 όπως στον ASCII
- 128...159 πρόσθετοι χαρακτήρες ελέγχου
- 160...255 ελληνικά και σχετικά σύμβολα

Πρότυπο Unicode

- Για την αναπαράσταση όλων των αλφαβήτων
 - Καλύπτει ιδεογράμματα, φωνητικές αναπαραστάσεις και διάφορα σύμβολα (~100.000 χαρακτήρες έχουν οριστεί)
 - Θεωρητικά μπορεί να καλύψει πάνω από 1 εκ. χαρακτήρες
- Κάθε χαρακτήρας αναπαρίσταται με έναν δυαδικό αριθμό (codepoint)
 - 0 έως 10FFFF
 - Χρειάζονται **περισσότερα από ένα bytes** για την αποθήκευση ενός τέτοιου αριθμού
 - Με περισσότερα από 1 bytes ανά χαρακτήρα τίθεται θέμα σειράς αποθήκευσης των bytes (little- ή big-endian)

Πρότυπο Unicode

- Το πρότυπο Unicode περιέχει επίσης
 - Πληροφορία ισοδύναμων ή παρόμοιων χαρακτήρων
 - Συνδυασμούς τόνων/διακριτικών και γραμμάτων
 - Οδηγίες για την ταξινόμηση των γραμμάτων ανά γλώσσα

Ελληνικά και Unicode

Greek and Coptic

03FF

	037	038	039	03A	03B	03C	03D	03E	03F
0			ι̇ 0390	Π 03A0	ϋ̇ 03B0	π 03C0	β 03D0	Ϡ 03E0	κ 03F0
1			Α 0391	Ρ 03A1	α 03B1	ρ 03C1	ϑ 03D1	ϡ 03E1	ρ 03F1
2			Β 0392		β 03B2	ς 03C2	Υ 03D2	Ϡ 03E2	Ϸ 03F2
3			Γ 0393	Σ 03A3	γ 03B3	σ 03C3	Υ̇ 03D3	Ϡ 03E3	ι̇ 03F3
4	΄ 0374	΄ 0384	Δ 0394	Τ 03A4	δ 03B4	τ 03C4	Υ̇ 03D4	Ϡ 03E4	Θ 03F4
5	΄ 0375	Ⲁ 0385	Ε 0395	Υ 03A5	ε 03B5	υ 03C5	φ 03D5	Ϡ 03E5	€ 03F5
6		Α̇ 0386	Ζ 0396	Φ 03A6	ζ 03B6	φ 03C6	ω 03D6	Ϡ 03E6	ε̇ 03F6

δεν φαίνεται όλος ο πίνακας

Κωδικοποίηση Unicode

- Το πρότυπο Unicode αναθέτει έναν αριθμό (codepoint) σε κάθε χαρακτήρα των αλφαβήτων που υποστηρίζει
 - Π.χ. ο λατινικός χαρακτήρας **L** αντιστοιχεί στον αριθμό **4C** και ο ελληνικός χαρακτήρας **ψ** στον αριθμό **3C8**
 - Οι γλώσσες προγραμματισμού που υποστηρίζουν Unicode χρησιμοποιούν **4 (ή 2) bytes** ανά χαρακτήρα στη μνήμη
- Κατά την αποθήκευση όμως σε αρχείο χρησιμοποιείται κάποιας μορφής κωδικοποίηση
 - Μετατρέπει τους αριθμούς Unicode σε μια σειρά από bytes με καθορισμένη **μορφή** και **σειρά**
 - Η αντίστροφη μετατροπή από τα bytes ενός αρχείου σε αριθμούς Unicode ονομάζεται **αποκωδικοποίηση**.

Κείμενο σε Unicode

- Παράδειγμα με 2 bytes ανά χαρακτήρα

	Γ	ε	ι	α		σ	ο	υ	!
δεκαδικό	915	949	953	945	32	963	959	965	33
δεκαεξαδικό	0393	03B5	03B9	03B1	0020	03C3	03BF	03C5	0021

Κωδικοποίηση big-endian

03	93	03	B5	03	B9	03	B1	00	20	03	C3	03	BF	03	C5	00	21
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Κωδικοποίηση little-endian

93	03	B5	03	B9	03	B1	03	20	00	C3	03	BF	03	C5	03	21	00
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Unicode σε κωδικοποίηση UTF-8

- Αναπαράσταση μεταβλητού μήκους

Unicode	Κωδικοποίηση UTF-8
00...7F	0xxxxxxx
80...7FF	110xxxxx 10xxxxxx
800...FFFF	1110xxxx 10xxxxxx 10xxxxxx
10000...10FFFF	11110xxx 10xxxxxx 10xxxxxx 10xxxxxx

- Το βασικό λατινικό αλφάβητο χρησιμοποιεί 1 byte ανά χαρακτήρα
 - Προς τα πίσω συμβατότητα με τον κώδικα ASCII
- Τα ελληνικά, 2 bytes
- Αλφάβητα Άπω Ανατολής, 3+ bytes