

1.2 Πόσα bits είναι αρκετά;

Με τους αριθμούς μπορούμε να αναπαραστήσουμε/κωδικοποιήσουμε διάφορα πράγματα. Για παράδειγμα, αν θέλαμε να αναπαραστήσουμε τις μέρες της εβδομάδας στη μνήμη του υπολογιστή, θα έπρεπε να επιλέξουμε κάποιον δυαδικό αριθμό για κάθε μέρα. Είναι φανερό ότι δε μπορούμε να πετύχουμε το στόχο μας αν έχουμε στη διάθεσή μας μόνο ένα bit το οποίο μπορεί να είναι 0 ή 1. Αν αντιστοιχίσουμε το 0 στην Κυριακή και το 1 στη Δευτέρα, δεν έχουμε άλλο σύμβολο για την Τρίτη.

0	00	Κυριακή
1	01	Δευτέρα
2	10	Τρίτη
3	11	Τετάρτη
4	?	Πέμπτη

Αναγκαστικά, θα χρειαστούμε και δεύτερο bit. Πάλι, όμως, έχουμε το ίδιο πρόβλημα. Με δυο bits όλοι οι πιθανοί συνδυασμοί είναι 4, οπότε μπορούμε να αναπαραστήσουμε μόνο τέσσερις μέρες. Άρα, θα χρειαστούμε και άλλα bits. Παρατηρήστε ότι κάθε συνδυασμός των 2 bits επαναλαμβάνεται μια φορά με το 1 μπροστά και μια με το 0. Άρα, κάθε φορά που προσθέτουμε 1 bit, οι πιθανοί συνδυασμοί διπλασιάζονται. Από 4 bits πάμε στα $2 \cdot 4 = 8$ bits. Οι μέρες είναι 7, άρα τα 3 bits αρκούν. Όπως φαίνεται δίπλα έχουμε αντιστοιχήσει σε κάθε μέρα μια ακολουθία δυαδικών ψηφίων (bits) και μας περισσεύει και μια θέση.

0	000	Κυριακή
1	001	Δευτέρα
2	010	Τρίτη
3	011	Τετάρτη
4	100	Πέμπτη
5	101	Παρασκευή
6	110	Σάββατο
7	111	

Μπορείτε να σκεφτείτε πόσα bits χρειαζόμαστε για να αναπαραστήσουμε τους μήνες του χρόνου;

Υπολογισμός πληθους Συνδιασμων

1 bit

Αρκούν τα 3 bits για 12 μήνες;

Άρα

Με 2 bits, μπορούμε να κωδικοποιήσουμε $4 = 2^2$ σύμβολα

Με 3 bits, μπορούμε να κωδικοποιήσουμε $8 = 2^3$ σύμβολα

Με 4 bits, μπορούμε να κωδικοποιήσουμε $16 = 2^4$ σύμβολα

Με 8 bits, μπορούμε να κωδικοποιήσουμε $256 = 2^8$ σύμβολα



Δραστηριότητα 1

1. Πόσα σύμβολα μπορούμε να κωδικοποιήσουμε α) με 16 bits, β) με 32 bits; γ) με 64 bits;
2. Πόσα bits χρειάζονται για την κωδικοποίηση όλων των ελληνικών γραμμάτων;
3. Στο παρακάτω πλαίσιο υπάρχουν συνοπτικές πληροφορίες για τον κώδικα Μορς. Πόσα bits χρειάζονται για την αναπαράσταση μηνυμάτων του κώδικα Μορς;
4. Πόσα bits χρειάζονται, αν θέλουμε να έχουμε στο ίδιο κείμενο αγγλικά, ελληνικά γράμματα, αριθμούς καθώς και τους ειδικούς χαρακτήρες: τελεία, κενό, ερωτηματικό και κόμμα;



Κώδικας Μορς (Morse code)

Ο κώδικας Μορς, μια μέθοδος μετάδοσης πληροφοριών, που χρησιμοποιεί μια σειρά από τελείες, παύλες και κενά, αναπτύχθηκε τη δεκαετία του 1830 από τον Samuel Morse, Αμερικανό καλλιτέχνη και εφευρέτη, και τον Alfred Vail, τον συνεργάτη του. Η εφεύρεση του ηλεκτρομαγνητικού τηλεγράφου και η ανάπτυξη του κώδικα Μορς επέτρεψαν την ασύρματη μεταφορά μηνυμάτων με τη μορφή κωδικών Μορς μέσω ραδιοκυμάτων, γνωστή ως ασύρματη τηλεγραφία (wireless telegraphy). Ο κώδικας Μορς παρέμεινε ένα σημαντικό μέσο ναυτικής επικοινωνίας μέχρι τα μέσα του 20^{ου} αιώνα, με τους χειριστές να μπορούν να μεταδίδουν μηνύματα με ρυθμούς έως και 200 λέξεις ανά λεπτό.

Στην Εικόνα 1.1. βλέπουμε την αναπαράσταση των γραμμάτων του αγγλικού αλφαβήτου. Από τις πιο διάσημες αναπαραστάσεις είναι αυτή του σήματος διάσωσης SOS: ••• — — — •••

A •—	J •— — —	S •••
B —•••	K —•—	T —
C —•—•	L —•••	U ••—
D —••	M — —	V •••—
E •	N —•	W •— —
F ••—•	O — — —	X —••—
G — —•	P •— —•	Y —•— —
H ••••	Q — —•—	Z — —••
I ••	R •—•	

Εικόνα 1.1. Κώδικας Μορς για το αγγλικό αλφάβητο

Κωδικας Μορς

1.3 bits & Bytes

Η πληροφορία μπορεί ν' αναπαρασταθεί με δυο τρόπους: αναλογικά και ψηφιακά. Η αναλογική πληροφορία παίρνει όλες τις δυνατές τιμές εντός ενός συγκεκριμένου εύρους, όπως ακριβώς συμβαίνει με τους πραγματικούς αριθμούς. Στην άλλη περίπτωση, μεταξύ των ακέραιων αριθμών 1 και 2 δεν υπάρχει άλλος αριθμός.

Γνωρίζουμε ότι κάθε σύμβολο πάνω στο πληκτρολόγιο, είτε είναι αριθμός, είτε γράμμα, είτε σημείο στίξης, αποθηκεύεται στη μνήμη του υπολογιστή ως μια ακολουθία από bits, δηλαδή δυαδικά ψηφία 0 ή 1.

Για ένα μεγάλο διάστημα, κάθε γράμμα, ψηφίο ή ειδικός χαρακτήρας αποθηκεύεται στη μνήμη του υπολογιστή ως μια οκτάδα από bits, γνωστή και ως Byte. Έτσι κάθε οκτάδα από bits στη μνήμη του υπολογιστή συνιστά 1 Byte.

8 bits = 1 Byte

1	1	1	1	1	0	1	1
---	---	---	---	---	---	---	---

Τάξη μεγέθους	Συμβολισμός	Bytes		Προσεγγιστικά
1 KiloByte	1 KB	2 ¹⁰ Bytes	1024 Bytes	1.000 Bytes
1 MegaByte	1 MB	2 ¹⁰ KB	2 ²⁰ Bytes	10 ⁶ Bytes
1 GigaByte	1 GB	2 ¹⁰ MB	2 ³⁰ Bytes	10 ⁹ Bytes
1 TeraByte	1 TB	2 ¹⁰ GB	2 ⁴⁰ Bytes	10 ¹² Bytes
1 PetaByte	1 PB	2 ¹⁰ TB	2 ⁵⁰ Bytes	10 ¹⁵ Bytes
1 ExaByte	1 EB	2 ¹⁰ PB	2 ⁶⁰ Bytes	10 ¹⁸ Bytes

Ο κώδικας ASCII σχεδιάστηκε για το αγγλικό αλφάβητο και δεν κάλυπτε σε ικανοποιητικό βαθμό γράμματα αλφαβήτων άλλων γλωσσών. Με την ανάπτυξη του διαδικτύου και του παγκόσμιου ιστού εμφανίστηκε η ανάγκη για αποτύπωση πολυγλωσσικών κειμένων. Γι' αυτό, αναπτύχθηκε ο κώδικας Unicode, στον οποίο κάθε σύμβολο χρησιμοποιεί από 8 έως και 32 bits, δηλαδή από 1 έως 4 Bytes. Στο πλαίσιο του βιβλίου, θεωρείται ότι κάθε σύμβολο κωδικοποιείται με 1 Byte = 8 bits για λόγους ευκολίας στους υπολογισμούς.



μετατροπή
χαρακτήρα σε
δυναδικο

ΠΙΝΑΚΑΣ ASCII							
Δυναδικός	Χαρακτήρας	Δυναδικός	Χαρακτήρας	Δυναδικός	Χαρακτήρας	Δυναδικός	Χαρακτήρας
00100000	Κενό	00111010	:	01010100	T	01101110	n
00100001	!	00111011	;	01010101	U	01101111	o
00100010	"	00111100	<	01010110	V	01110000	p
00100011	#	00111101	=	01010111	W	01110001	q
00100100	\$	00111110	>	01011000	X	01110010	r
00100101	%	00111111	?	01011001	Y	01110011	s
00100110	&	01000000	@	01011010	Z	01110100	t
00100111	'	01000001	A	01011011	[	01110101	u
00101000	(	01000010	B	01011100	\	01110110	v
00101001	)	01000011	C	01011101	]	01110111	y
00101010	*	01000100	D	01011110	^	01110111	w
00101011	+	01000101	E	01011111	_	01111000	x
00101100	,	01000110	F	01100000	`	01111010	z
00101101	-	01000111	G	01100001	a	01111011	{
00101110	.	01001000	H	01100010	b	01111100	
00101111	/	01001001	I	01100011	c	01111101	}
00110000	0	01001010	J	01100100	d	01111110	~
00110001	1	01001011	K	01100101	e	01111111	DEL
00110010	2	01001100	L	01100110	f		
00110011	3	01001101	M	01100111	g		
00110100	4	01001110	N	01101000	h		
00110101	5	01001111	O	01101001	i		
00110110	6	01010000	P	01101010	j		
00110111	7	01010001	Q	01101011	k		
00111000	8	01010010	R	01101100	l		
00111001	9	01010011	S	01101101	m		



Δραστηριότητα 2

1. Πόσα Bytes χωράει ένας σκληρός δίσκος με χωρητικότητα 500 GB;
2. Πόσα γράμματα μπορούν να αποθηκευτούν σε ένα σκληρό δίσκο με χωρητικότητα 256 GB;
3. Ένα κείμενο με 2.000 Bytes (χαρακτήρες), πόσα bits (ψηφία) είναι;
4. Ένα κείμενο 80.000 bits, πόσα Bytes είναι;
5. Πόσα MB είναι τα 10.000 KB;
6. Πόσα GB είναι τα 5.000 MB;
7. Ποια είναι η τάξη μεγέθους της πληροφορίας που υπάρχει σήμερα διαθέσιμη στον παγκόσμιο ιστό;



Παράδειγμα 1

Ένα βιβλίο έχει 500 σελίδες, κάθε σελίδα έχει 40 γραμμές και κάθε γραμμή έχει 50 χαρακτήρες. Επίσης το βιβλίο περιέχει εικόνες συνολικού μεγέθους 3 MB. Πόσα τέτοια βιβλία χωράνε σε ένα USB stick 32 GB και πόσα σε έναν σκληρό δίσκο 3,2 TB;

Απάντηση

Αρχικά θα πρέπει να βρούμε πόσα Bytes χρειάζεται ένα βιβλίο για να αποθηκευτεί στη μνήμη του υπολογιστή. Έχουμε ότι:

1 χαρακτήρας είναι 1 Byte

1 γραμμή έχει 50 χαρακτήρες, άρα 50 Bytes

1 σελίδα έχει 40 γραμμές, άρα $40 \times 50 \text{ Bytes} = 2.000 \text{ Bytes} = 2 \text{ KB}$

1 βιβλίο έχει 500 σελίδες, άρα $500 \times 2 \text{ KB} = 1.000 \text{ KB} = 1 \text{ MB}$

Συνολικά ένα βιβλίο έχει μέγεθος 1 MB (κείμενο) + 3MB (εικόνες) = 4 MB.

Σε 32 GB, πόσα βιβλία μπορούν να αποθηκευτούν;

$$\frac{32 \text{ GB}}{4 \text{ MB}} = \frac{32.000 \text{ MB}}{4 \text{ MB}} = 8.000 \text{ βιβλία}$$

Άρα, σε ένα USB Stick μπορούν να αποθηκευτούν περίπου **8.000 βιβλία**.

Ένας σκληρός δίσκος $3,2 \text{ TB} = 3,2 \cdot 1.000 \text{ GB} = 3.200 \text{ GB} = 100 \cdot 32 \text{ GB}$. Άρα, αφού σε 32GB μπορούν να αποθηκευτούν 8.000 βιβλία, τότε στον σκληρό δίσκο αποθηκεύονται $100 \cdot 8000 = \mathbf{800.000 \text{ βιβλία}}$.

Σημείωση: Παρατηρήστε ότι δε μπορούμε να διαιρέσουμε GB με MB. Θα έπρεπε να τα μετατρέψουμε σε κάτι κοινό, όπως GigaBytes. Θα μπορούσαμε να μετατρέψουμε και τα δύο σε Bytes, αλλά τότε θα είχαμε μεγάλα νούμερα με πολλά μηδενικά. Εκμεταλλευτήκαμε το γεγονός ότι $1 \text{ TB} \cong 1.000 \text{ GB}$.