

Εισαγωγή στους Η/Υ (Θεωρία)

Επανάληψη I

*Στο σύνολο της ύλης
(θεωρία & προγραμματισμός)*

Δίκτυα & Συστήματα Αρίθμησης

Handwritten calculation:

$$\begin{array}{r|l} 192 & 2 \\ \hline 96 & \emptyset \\ \hline 48 & \emptyset \end{array}$$

Handwritten: $\emptyset \emptyset$ with an arrow pointing left.

Ένας υπολογιστής έχει διεύθυνση IPv4: **192.168.26.88** και ο Router στο δίκτυο του υπολογιστή έχει (εκφρασμένη στο δεκαεξαδικό) την: **C0.A8.18.55**

Τι ρύθμιση πρέπει να κάνει ο τεχνικός στη **μάσκα δικτύου** του Router ώστε να λειτουργήσει σωστά το δίκτυο και να χωρούν σε αυτό όσο **περισσότερες** συσκευές γίνεται;

Handwritten: Μάσκα 22

H/Y	192	168	26	88
	11000000	101010000000	0110100101100000	
Router	C0	A8	18	55
	11000000	101010000000	011000001010101	
Μάσκα	255	255	252	
	11111111	11111111	1111110000000000	

Handwritten: Most Part N-bits

Handwritten: $2^N - 2$

Handwritten: 608 ΚΕΥ.

Handwritten: Σύμβολο

Handwritten: $2^{10} - 2$

Handwritten: \emptyset and \emptyset with arrows pointing to the 26 and 88 columns.

Handwritten: 10

Παραστάσεις C

Δεδομένου ότι ο κώδικάς μας ξεκινάει με:

```
double a, b, c, d;
```

```
long i, j, m;
```

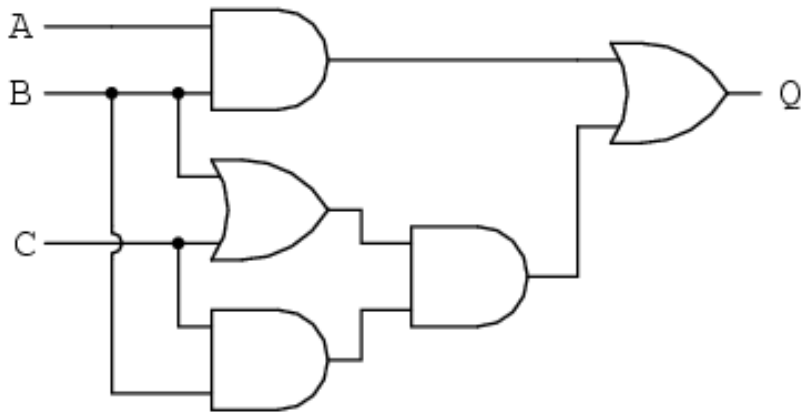
```
bool X, Y, Z;
```

Γράψτε τις παρακάτω **εντολές** ώστε να:

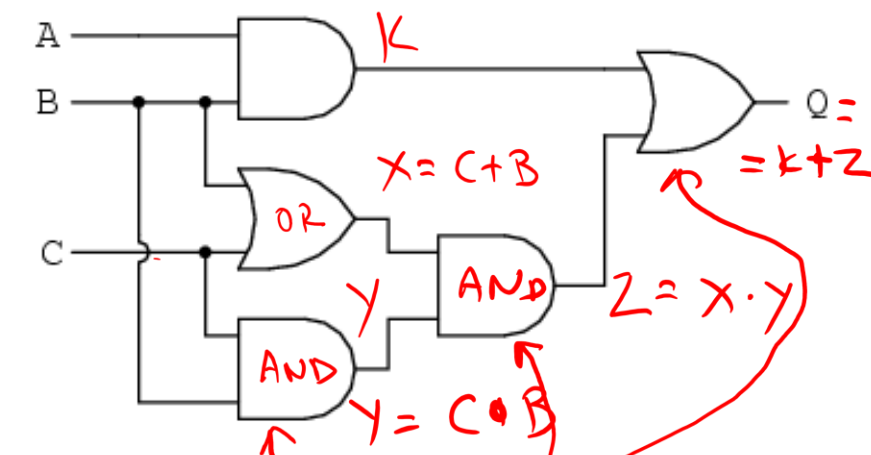
Αποθηκευτεί στην a η μέση τιμή των i, j, m	$a = \frac{i + j + m}{3.0};$ <p><i>int</i> (under i, j, m) <i>int</i> → double / float (under 3.0)</p>
Αποθηκευτεί στην a η τιμή της $\frac{b-c}{b \cdot d}$	$a = (b - c) / (b * d);$ <p><i>[1/b / d]</i></p>
Αποθηκευτεί στην X το κατά πόσο οι Y και Z είναι και οι δύο αληθείς ή και οι δύο ψευδείς	$X = (Y \&\& Z) \parallel (!Y \&\& !Z);$ <p><i>X = (Y == Z);</i></p>

Λογικά κυκλώματα & C

Δίνεται το παρακάτω λογικό διάγραμμα με εισόδους τις λογικές μεταβλητές A, B, C και με έξοδο τη λογική μεταβλητή Q. Ζητούνται:



1. Να συμπληρωθεί ο πίνακας αληθείας του κυκλώματος (συμφέρει να χρησιμοποιήσετε τις κενές στήλες για ενδιάμεσες ποσότητες/παραστάσεις).
2. Να γραφεί η αντίστοιχη λογική συνάρτηση Q
3. Να γραφεί σε C η ίδια παράσταση θεωρώντας ότι οι Q,A,B,C είναι λογικές μεταβλητές



A	B	C	X	Y	Z	K	Q
0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	0
0	1	0	1	0	0	0	0
0	1	1	1	1	1	0	1
1	0	0	0	0	0	0	0
1	0	1	1	0	0	0	0
1	1	0	1	0	0	1	1
1	1	1	1	1	1	1	1

$$Q = (B + C) \cdot (C \cdot B) + (A \cdot B)$$

$$Q = (B \parallel C) \&\&(C \&\&B) \parallel (A \&\&B);$$

Λογικά κυκλώματα & C

Συμπληρώστε τον παρακάτω πίνακα αλήθειας ώστε το Q να δείχνει κατά πόσο ο αριθμός ABC₍₂₎ είναι πολλαπλάσιο του 4. Γράψτε την αντίστοιχη λογική παράσταση, καθώς και την αντίστοιχη παράσταση σε C. Τέλος δημιουργείστε το αντίστοιχο λογικό κύκλωμα:

$$Q = \overline{A+B} \quad Q = \overline{(A \parallel B)}$$

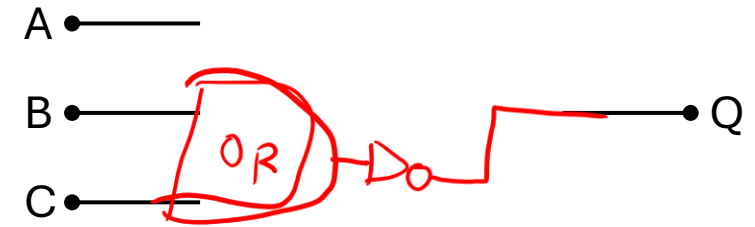
	A	B	C	Q
0	0	0	0	1
1	0	0	1	0
2	0	1	0	0
3	0	1	1	0
4	1	0	0	1
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

Επειδή είναι ίδια
τα B & C με OR

εχω 0 μόνο στις

αφαιρούμε ως που με ενδιαφέρουν.

Αρα με το NOT (\neg) παίρνω 1 (true).



Διόρθωση κώδικα C

Ο παρακάτω κώδικας C υπολογίζει το άθροισμα των στοιχείων κάθε στήλης πίνακα, με την ίδια λογική που το κάνει η συνάρτηση sum της Matlab. Έχει όμως τρία συντακτικά (όχι λογικά) λάθη, διορθώστε τα πάνω στον κώδικα.

```
double *matlabSum(int R, int C, double **A) {  
    if (R < 1 || C < 1) {  
        return NULL;  
    }  
    double *result = (double *) malloc(sizeof(double) * C);  
    if (result == NULL) {  
        return NULL;  
    }  
    for (int c = 0; c < C; ++c) {  
        result[c] = 0;  
        for (int r = 0; r < R; ++r) {  
            result[c] += A[r][c];  
        }  
    }  
    return result;  
}
```

double result = ϕ ;

Κατόπιν, τροποποιείστε τον
ώστε να υπολογίζει ένα
από τα ακόλουθα:

- (α)** το γινόμενο των
στοιχείων κάθε στήλης
- (β)** το άθροισμα όλων των
στοιχείων του πίνακα
- (γ)** το άθροισμα των
θετικών στοιχείων κάθε
στήλης
- (δ)** το μέγιστο των
στοιχείων κάθε στήλης