



Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης

Ψηφιακές Επικοινωνίες
7ο Εργαστήριο

Όνομα: Χρήστος-Αλέξανδρος Τσιγγιρόπουλος
ΑΕΜ: 3872

[Εκφώνηση Εργαστηρίου \(.docs\)](#)

Γλώσσα Προγραμματισμού [Octave](#)

1) Περιγράψτε την έννοια της ισοτιμίας και δημιουργήστε κώδικα, που να υπολογίζει άρτια ισοτιμία για ένα μήνυμα των 14 bits και να προσαρτά σε αυτό ένα bit ισοτιμίας. Τρέξτε τον κώδικά σας για το μήνυμα 10001101011000.

Απάντηση:

Με λίγα λόγια ο έλεγχος της ισοτιμίας (άρτιας ή περιττής) παίρνει ένα μήνυμα και προσθέτει στο τέλος ένα bit . Η τιμή του bit αυτού εξαρτάται από το πλήθος των bits που έχουν τιμή “1”. Στην άρτια ισοτιμία το πλήθος των “1” μαζί με την προσθήκη του bit αυτού πρέπει να είναι άρτιος αριθμός , ενώ στην περιττή , περιττός.

Παράδειγμα:

Στο παράδειγμα μας το αρχικό μήνυμα είναι 10001101011000
Αφού έχουμε άρτια ισοτιμία και το πλήθος των 1 είναι άρτιος το bit που θα προστεθεί θα έχει την τιμή 0. Οπότε το τελικό μήνυμα θα είναι το 100011010110000 .

► Το αρχείο της πρώτης άσκησης μπορεί να βρεθεί και [εδώ](#).

2) Δημιουργήστε κώδικα που να ελέγχει αν υπάρχει σφάλμα σε ένα bit για το μήνυμα των 14 bits που συνοδεύεται από ένα bit άρτιας ισοτιμίας. Τρέξτε τον κώδικά σας για να ελέγξετε το αν έχει φτάσει χωρίς σφάλμα το μήνυμα 10001101011000 του προηγούμενου σκέλους της άσκησης (και το bit ισοτιμίας που το συνοδεύει) στις εξής περιπτώσεις:

- i. Αφίξη στον αποδέκτη χωρίς σφάλμα.
- ii. Αφίξη στον αποδέκτη με ένα σφάλμα (στο 6^ο bit, με αρίθμηση των bit μετρώντας από αριστερά).
- iii. Αφίξη στον αποδέκτη με δύο σφάλματα (στο 6^ο bit και στο 12^ο bit, με αρίθμηση των bit μετρώντας από αριστερά).

Απάντηση:

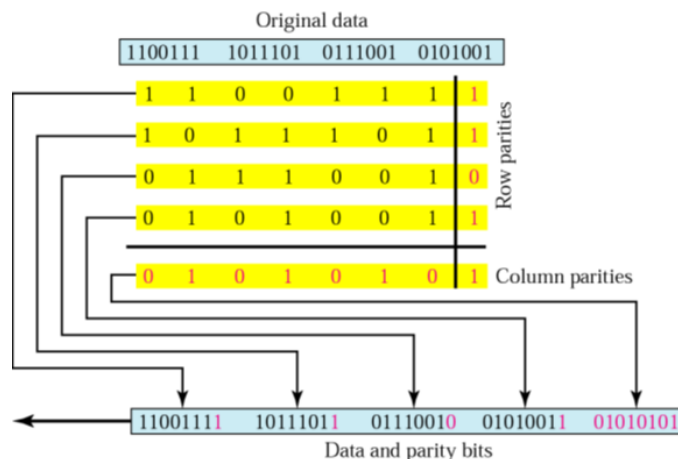
i. Αποτέλεσμα του κώδικα με είσοδο 100011010110000 είναι
ότι δεν υπάρχει σφάλμα (**Σωστά**)

ii. Εδώ η είσοδος είναι η 10001010110000 (σφάλμα 6ου bit)
και το πρόγραμμα βγάζει ότι υπάρχει σφάλμα στο μήνυμα (**Σωστά**)

iii. Τέλος με είσοδο 10001010111000 (σφάλμα 6ου & 12ου bit)
το πρόγραμμα βγάζει ότι δεν υπάρχει σφάλμα (**Λάθος**). Αυτό συμβαίνει
γιατί αλλάζοντας 2 bit η ισοτιμία αρχίζει να ξανά ισχύει οπότε το
πρόγραμμα δεν μπορεί να καταλάβει ότι υπάρχει λάθος.

► Το αρχείο της δεύτερης άσκησης μπορεί να βρεθεί και [εδώ](#).

3) Δημιουργήστε κώδικα που να διαιρεί ένα μήνυμα των 28 bits σε 4 γραμμές των 7 bits (όπως φαίνεται στο σχήμα), να υπολογίζει τα bits άρτιας δισδιάστατης ισοτιμίας για κάθε γραμμή και κάθε στήλη και να τα προσαρτά σε αυτές. Τρέξτε τον κώδικά σας για το συγκεκριμένο μήνυμα που φαίνεται στο παραπάνω σχήμα.

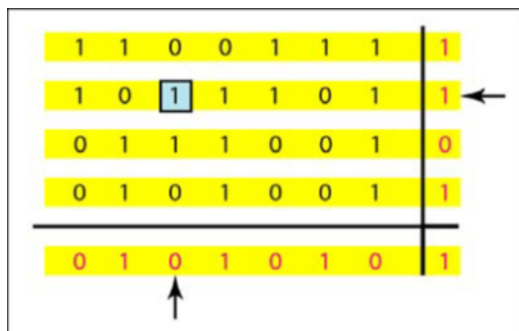


Απάντηση:

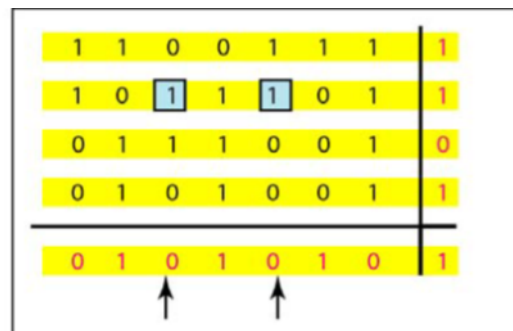
Το 1100111 1011101 0111001 0101001 είναι το original message αυτής της άσκησης. Το πρόγραμμα υπολογίζει τα bits άρτιας διαστάσεως ισοτιμίας και εκτυπώνει το αποτέλεσμα
11001111 10111011 01110010 01010011 01010101

- Το αρχείο της τρίτης άσκησης μπορεί να βρεθεί και [εδώ](#).

- 4) Δημιουργείστε κώδικα που να ανιχνεύει σφάλματα στον πίνακα δύο διαστάσεων (αποτελούμενο από bits μηνύματος και bits ισοτιμίας) που παράγεται από τον κώδικα του 3^{ου} σκέλους της άσκησης.
Τρέξτε τον κώδικά σας για το συγκεκριμένο μήνυμα του 3^{ου} σκέλους της άσκησης και για τις εξής περιπτώσεις:
- Ο πίνακας φτάνει χωρίς σφάλματα
 - Ο πίνακας φτάνει με σφάλματα σε ένα bit (στο συγκεκριμένο bit που φαίνεται με γαλάζιο χρώμα στο σχήμα b).
 - Ο πίνακας φτάνει με σφάλμα σε δύο bits (στα συγκεκριμένα bits που φαίνονται με γαλάζιο χρώμα στο σχήμα c).



b. One error affects two parities



c. Two errors affect two parities

Απάντηση:

- Αποτέλεσμα του του προηγούμενου ερωτήματος είναι ο πίνακας έχει την τιμή 11001111 10111011 01110010 01010011 01010101 . Αποτέλεσμα αυτού είναι ότι το μήνυμα αυτό δεν έχει λάθος. (Σωστά)
- Ο πίνακας 11001111 10011011 01110010 01010011 01010101 είναι ο πίνακας που περιγράφεται στο σχήμα b. Το πρόγραμμα με αυτή την είσοδο έχει έξοδο το μήνυμα της ύπαρξης λάθους . (Σωστά)
- Ο πίνακας 11001111 10010011 01110010 01010011 01010101 είναι ο πίνακας που περιγράφεται στο σχήμα c. Το πρόγραμμα με αυτή την είσοδο εξαγει πάλι το μήνυμα της ύπαρξης λάθους. (Σωστά)

► Το αρχείο της τέταρτης άσκησης μπορεί να βρεθεί και [εδώ](#).

5) Υπάρχουν κάποια πρότυπα σφάλματος που δεν ανιχνεύονται στη δισδιάστατη ισοτιμία;

Αν ναι, ποια είναι αυτά;

Απάντηση:

1	1	0	0	1	1	1	1
1	0	1	1	1	0	1	1
0	1	1	1	0	0	1	0
0	1	0	1	0	0	1	1
0	1	0	1	0	1	0	1

Four errors that affect zero parities

Υπάρχουν πρότυπα σφάλματος που δεν μπορούν να ανιχνευτούν μέσα από τη δισδιάστατη ισοτιμία, όπως με τις αλλαγές που παρουσιάζονται στην παραπάνω εικόνα. Μετά τις αλλαγές θα έχουμε τον πίνακα στα δεξιά μας, ο οποίος αν εισαχθεί σαν είσοδος στο πρόγραμμα, θα έχουμε ως αποτέλεσμα ότι το «μήνυμα» δεν έχει λάθος. Γενικά, το πρότυπο σφάλματος που δεν μπορεί να ανιχνευτεί είναι όταν αλλάζουν 4 στοιχεία που σχηματίζουν τετράγωνο και έχουν την ίδια τιμή. Αυτό γιατί, με

0	1	0	0	1	1	0	1
0	0	1	1	1	0	0	1
0	1	1	1	0	0	1	0
0	1	0	1	0	0	1	1
0	1	0	1	0	1	0	1

Four errors that affect zero parities

1	1	0	0	1	1	1	1
1	0	1	1	1	0	1	1
0	1	1	1	0	0	1	0
0	1	0	1	0	0	1	1
0	1	0	1	0	1	0	1

Four errors that affect zero parities

αυτές τις αλλαγές η ισοτιμία παραμένει ίδια οπότε ο αλγόριθμος δεν μπορεί να βρεί την ύπαρξη λάθους που δημιουργήθηκε κατά την μεταφορά του μηνύματος. Ένα δεύτερο παράδειγμα είναι αυτό με την αλλαγή των στοιχείων [3,1], [3,6], [4,1], [4,6] που παρουσιάζεται στο διπλανό πίνακα.

► Το αρχείο της πέμπτης άσκησης μπορεί να βρεθεί και [εδώ](#).