

ΑΡΙΣΤΟΤΕΛΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΟΝΙΚΗΣ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΜΑΘΗΜΑ: ΨΗΦΙΑΚΕΣ ΕΠΙΚΟΙΝΩΝΙΕΣ

ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΗ ΕΡΓΑΣΙΑ: ΥΛΟΠΟΙΗΣΗ CRC

ΟΝΟΜΑΤΕΠΩΝΥΜΟ: ΔΗΜΗΤΡΙΟΣ ΠΑΝΤΕΛΕΗΜΩΝ ΓΙΑΚΑΤΟΣ

ΑΕΜ: 3061



Περιεχόμενα

Εισαγωγή.....	3
Περιγραφή προγράμματος σε JAVA	3
Παράδειγμα	4
Υπολογισμός ποσοστών για $k = 10$, $P = 11010$ και $E = 0.001$ για 10000 μηνύματα.....	5
Οδηγίες εκτέλεσης jar αρχείου.....	5

Εισαγωγή

Στα πλαίσια του μαθήματος Ψηφιακές Επικοινωνίες καλούμαστε να υλοποιήσουμε τον αλγόριθμο ανίχνευσης λαθών CRC. Η υλοποίηση του προαναφερόμενου αλγορίθμου έγινε στη γλώσσα προγραμματισμού JAVA.

Περιγραφή προγράμματος σε JAVA

Το πρόγραμμα περιλαμβάνει τέσσερις κλάσεις (Main, Transceiver, BitErrorRate, Receiver). Η αρχική κλάση είναι η Main, η οποία περιλαμβάνει ένα στοιχειώδες User Interface ώστε ο χρήστης να μπορεί να δίνει στο πρόγραμμα τον προκαθορισμένο αριθμό P των $n+1$ bits, στον οποίο το πρώτο και το τελευταίο bit πρέπει να είναι 1. Σε περίπτωση που ο χρήστης δώσει ένα αριθμό όπου δεν ικανοποιεί την συνθήκη (το πρώτο και το τελευταίο bit να είναι 1) τότε το πρόγραμμα ξαναζητάει από τον χρήστη να ξαναδώσει τον αριθμό P . Στη συνέχεια το πρόγραμμα ζητάει από το χρήστη να δώσει το μέγεθος k bits του μηνύματος που θέλει να μεταδώσει. Το μέγεθος αυτό είναι αναγκαίο να είναι μεγαλύτερο ή ίσο από το μέγεθος του αριθμού P που έχει δώσει προηγούμενος. Αν δεν ικανοποιείται η συνθήκη τότε το πρόγραμμα ζητάει να ξαναδοθεί ο αριθμός των bits. Έπειτα ο χρήστης πληκτρολογεί τη πιθανότητα του σφάλματος E ($0.0 \leq E \leq 1.0$) και μετά δίνει το πλήθος των μηνυμάτων που θέλει να δημιουργηθούν και να σταλούν. Αμέσως μετά το πρόγραμμα για το πλήθος των μηνυμάτων που έχει δώσει ο χρήστης δημιουργεί τα μηνύματα, τα στέλνει στο κανάλι θορύβου και μετά τα στέλνει στον μεταδότη όπου ελέγχει αν το μήνυμα που έλαβε περιέχει σφάλμα ή όχι. Τέλος, η κλάση εκτυπώνει το ποσοστό των μηνυμάτων που φθάνουν με σφάλμα στον αποδέκτη, το ποσοστό των μηνυμάτων που ανιχνεύονται ως εσφαλμένα από το CRC και το ποσοστό των μηνυμάτων που φθάνουν με σφάλμα στο αποδέκτη και δεν ανιχνεύονται από το CRC, τα οποία υπολογίζονται από τη κλάση Receiver.

Η κλάση Transceiver δέχεται ως όρισμα τον προκαθορισμένο αριθμό P των $n+1$ bits και δημιουργεί ένα τυχαίο μήνυμα μεγέθους k . Αφού δημιουργήσει το μήνυμα τότε η κλάση προσθέτει στο τέλος του n μηδενικά, όπου n είναι ο αριθμός των bits του P των $n+1$ bits, και προκύπτει ο αριθμός $2^n M$. Ύστερα, διαιρούμαι τον αριθμό $2^n M$ με τον αριθμό P και από το υπόλοιπο της διαίρεσης προκύπτει ο αριθμός F . Αυτός ο αριθμός F προστίθεται στο τέλος του μηνύματος $2^n M$ και προκύπτει ο αριθμός T .

Η κλάση BitErrorRate δέχεται τον αριθμό T που προέκυψε από τη κλάση Transceiver και ανάλογα με την πιθανότητα σφάλματος που έχει δώσει ο χρήστης αλλοιώνει κάθε bit του μηνύματος. Με λίγα λόγια η κλάση δημιουργεί για κάθε ένα bit του T ένα αριθμό από το 0.0 μέχρι και το 1.0 και αν αυτός ο αριθμός είναι μικρότερος ή ίσος με τη πιθανότητα σφάλματος τότε αλλάζει το bit του μηνύματος από 0 σε 1 και από 1 σε 0.

Η κλάση Receiver δέχεται το μήνυμα που πιθανό να έχει αλλοιωθεί από τη κλάση BitErrorRate, δηλαδή το θόρυβο. Έπειτα διαιρεί τον αριθμό του μηνύματος με τον αριθμό P και αν το υπόλοιπο της διαίρεσης προκύψει 0 τότε το μήνυμα δεν έχει σφάλμα (δεν έχει αλλοιωθεί από το θόρυβο) ή έχει αλλοιωθεί από το θόρυβο και δεν έχει ανιχνευθεί από το CRC. Αν το υπόλοιπο είναι 1, τότε αυτό σημαίνει ότι ο ανιχνευτής λαθών CRC έχει εντοπίσει το σφάλμα στο μήνυμα.

Σημείωση: Στον κώδικα του προγράμματος υπάρχουν σχόλια που εξηγούν λεπτομερώς το κάθε μέρος του κώδικα.

Παράδειγμα

Σε αυτή τη ενότητα θα εξηγήσουμε το πως προκύπτει μαθηματικά η διαδικασία του αλγορίθμου ανίχνευσης λαθών CRC. Αρχικά έχουμε τον αριθμό $P = 110101$, τη πιθανότητα σφάλματος $E = 10^{-3} = 0.001$ και το μήκος του μηνύματος που θέλουμε να δημιουργεί $k = 10$. Μετά την εκτέλεση της κλάσης Transceiver στο πρόγραμμα ως υποθέσουμε ότι δημιουργείται τυχαία το μήνυμα $M = 1010001101$. Μετά στο μήνυμα M θα προστεθούν στο τέλος n μηδενικά, όπου n είναι ο αριθμός των $n + 1$ bits του P .

Ο αριθμός P έχει 6 bits:

$$n + 1 = 6$$

$$n = 6 - 1$$

$$n = 5$$

Άρα στο μήνυμα M θα προστεθούν στο τέλος πέντε μηδενικά:

$$2^n M = 2^5 M = 101000110100000$$

Μετά τη δημιουργία του αριθμού $2^5 M$ ξεκινάει η διαίρεση του με τον αριθμό P (ουσιαστικά εκτελείται κάθε φορά η πράξη XOR):

$2^5 M$	1	0	1	0	0	0	1	1	0	1	0	0	0	0	0
P	1	1	0	1	0	1									
	1	1	1	0	1	1									
	1	1	0	1	0	1									
		1	1	1	0	1	0								
		1	1	0	1	0	1								
			1	1	1	1	1	0							
			1	1	0	1	0	1							
				1	0	1	1	0	0						
				1	1	0	1	0	1						
					1	1	0	0	1	0					
					1	1	0	1	0	1					
						1	1	1	1	0					

Το υπόλοιπο της διαίρεσης που προέκυψε είναι ο αριθμός R τον οποίο τον προσθέτουμε στον αριθμό $2^5 M$ και προκύπτει ο αριθμός T :

$$T = 2^5 M + R = 101000110100000 + 1110 = 101000110101110$$

Στη συνέχεια ο αριθμός T πηγαίνει στη κλάση BitErrorRate, δηλαδή πηγαίνει σε ένα κανάλι θορύβου που ανάλογα με την πιθανότητα σφάλματος αλλοιώνει τα bits του αριθμού T . Ας υποθέσουμε ότι για κάθε bit του αριθμού T προέκυψε σφάλμα που είναι μεγαλύτερο της πιθανότητας σφάλματος $E = 0.001$ και άρα ο αριθμός T πήγε αναλλοίωτος στην κλάση Receiver, ουσιαστικά από αυτό το σημείο γνωρίζουμε ότι το μήνυμα που μεταδόθηκε δεν περιέχει σφάλμα. Έπειτα στη κλάση Receiver θα γίνει διαίρεση του αριθμού T με τον αριθμό P (ουσιαστικά εκτελείται κάθε φορά η πράξη XOR) για να δούμε αν το μήνυμα περιέχει σφάλμα:

$2^5 M$	1	0	1	0	0	0	1	1	0	1	0	1	1	1	0
P	1	1	0	1	0	1									
		1	1	1	0	1	1								
		1	1	0	1	0	1								
			1	1	1	0	1	0							
			1	1	0	1	0	1							
				1	1	1	1	1	0						
				1	1	0	1	0	1						
					1	1	1	1	1	0					
					1	1	0	1	0	1					
						1	0	1	1	1	1				
						1	1	0	1	0	1				
							1	1	0	1	0	1			
								1	1	0	1	0	1		
									1	1	0	1	0	1	
										1	1	0	1	0	1
											1	1	0	1	0
												1	1	0	1
													1	1	0
														1	0
															0

Το υπόλοιπο της διαίρεσης που προέκυψε $R = 0$ δηλώνει ότι το μήνυμα που μεταδόθηκε δεν περιείχε σφάλμα. Τέλος, αν το μήνυμα είχε σφάλμα δηλαδή είχε αλλοιωθεί έστω και ένα bit τότε το υπόλοιπο της παραπάνω διαίρεσης θα ήταν 1.

Υπολογισμός ποσοστών για $k = 10$, $P = 11010$ και $E = 0.001$ για 10000 μηνύματα

- Το ποσοστό των μηνυμάτων που φθάνουν με σφάλμα στον αποδέκτη: 1.61%
- Το ποσοστό των μηνυμάτων που ανιχνεύονται ως εσφαλμένα από το CRC: 1.61%
- Το ποσοστό των μηνυμάτων που φθάνουν με σφάλμα στο αποδέκτη και δεν ανιχνεύονται από το CRC: 0.0%

Σημείωση: Κάθε φορά η πιθανότητα μπορεί να αλλάζει γιατί το σφάλμα που συγκρίνεται με τη πιθανότητα σφάλματος E , δημιουργείται με τυχαίο τρόπο.

Οδηγίες εκτέλεσης jar αρχείου

Το αρχείο jar εκτελείται από command line.

Για την εκτέλεση του jar εκτελέστε τα παρακάτω βήματα:

1. Ανοίξτε το command line.
2. Προηγηθείτε στο directory που βρίσκεται το αρχείο jar.
3. Γράψτε και εκτελέστε την εντολή: `java -jar CRC.jar`