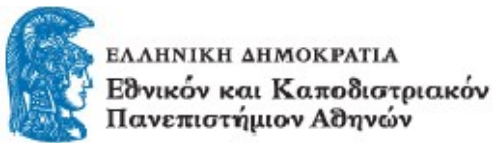


ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΤΜΗΜΑ ΜΑΘΗΜΑΤΙΚΩΝ

ΣΥΝΘΕΣΗ ΚΑΙ ΑΞΙΟΛΟΓΗΣΗ ΓΕΝΝΗΤΡΙΏΝ ΨΕΥΔΟΤΥΧΑΙΩΝ ΑΚΟΛΟΥΘΙΩΝ

SYNTHESIS AND EVALUATION OF PSEUDORANDOM NUMBER GENERATOS

ΒΑΣΙΛΟΓΙΑΝΝΗΣ ΓΕΩΡΓΙΟΣ

150303

Διπλωματική Διατριβή για την απόκτηση του Μεταπτυχιακού Τίτλου Σπουδών στα Εφαρμοσμένα Μαθηματικά

ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ

ΜΠΑΡΔΗΣ ΝΙΚΟΛΑΟΣ - Αναπληρωτής Καθηγητής ΣΣΕ

ΑΘΗΝΑ

2019

Περίληψη

Στην παρούσα διπλωματική εργασία, αναλύονται οι γεννήτριες ψευδοτυχαίων ακολουθιών που έχουν εφαρμογή στα σύγχρονα κρυπτογραφικά συστήματα. Αναπτύσσεται το μαθηματικό υπόβαθρο που βασίζεται η θεωρία των γεννητριών των ψευδοτυχαίων ακολουθιών καθώς και τα βασικά συστήματα σύνθεσης τους. Γίνεται λεπτομερειακή παρουσιάση και ανάλυση σύγχρονων κριτηρίων αξιολόγησης των γεννητριών ψευδοτυχαίων ακολουθιών λαμβάνοντας υπόψη τα διεθνή πρότυπα.

Γίνεται κατηγοριοποίηση των ψευδοτυχαίων ακολουθιών και πραγματοποιείται μια λεπτομερειακή καταγραφή των σύγχρονων γεννητριών ψευδοτυχαίων ακολουθιών που χρησιμοποιούνται σε κρυπτογραφικούς αλγορίθμους ροής. Δίνεται έμφαση στις γεννήτριες των ψευδοτυχαίων ακολουθιών που βασίζονται στους κυλιόμενους καταχωρητές με γραμμική ανάδραση (LFSR) οι οποίες κατέχουν το μεγαλύτερο μέρος των εφαρμογών. Στη συνέχεια παρουσιάζονται τεχνικές βελτίωσης των γεννητριών μέσω της εισαγωγής μη γραμμικών πράξεων. Ακολούθως αναλύονται γνωστές μη γραμμικές ΓΨΑ, οι οποίες αποτελεί βασικό αντιπρόσωπο στον τομέα της.

Τέλος πραγματοποιείται λεπτομερής ανάλυση των γεννητριών που αναπτύχθηκαν στη μελέτη καθώς και κριτηρίων αξιολόγησης τους.

Abstract

Η παρούσα Διπλωματική Διατριβή, εκπονήθηκε για της ανάγκες της απόκτησης του Μεταπτυχιακού Τίτλου Σπουδών της Κατεύθυνσης των Εφαρμοσμένων Μαθηματικών του ΠΜΣ ¨Μαθηματικά¨ του τμήματος Μαθηματικών, της Σχολής Θετικών Επιστημών, του Εθνικού και Καποδιστριακού Πανεπιστημίου Αθηνών.

ΤΡΙΜΕΛΗΣ ΕΠΙΤΡΟΠΗ

ΕΠΙΒΛΕΠΩΝ

ΜΠΑΡΔΗΣ ΝΙΚΟΛΑΟΣ - Αναπληρωτής Καθηγητής ΣΣΕ

ΜΕΛΗ

ΔΡΑΚΟΠΟΥΛΟΣ ΜΙΧΑΗΛ - Επίκουρος Καθηγητής τμήματος Μαθηματικών ΕΚΠΑ

ΤΡΕΒΕΖΑΣ ΣΑΜΗΣ - Λέκτορας τμήματος Μαθηματικών ΕΚΠΑ

Ευχαριστίες

Περιεχόμενα

Εισαγωγή

Αν και η ύπαρξη τυχαίων αριθμών έχει αποδειχθεί από το αξίωμα του Kolmogorov για την πιθανότητα (Neveu, Jacques: Mathematicial foundations of the calculus of probability. Vol.1. San Francisco: Holden-day, 1965), η αναζήτηση μεθόδων παραγωγής τέτοιων αριθμών αποτελεί μια από τις μεγαλύτερες προκλήσεις των θετικών επιστημών. Στην πάροδο των χρόνων οι τομείς της ζωής του ανθρώπου που χρησιμοποιούν εφαρμογές, οι οποίες βασίζονται ή απλά χρησιμοποιούν μεγάλο πλήθος αριθμών που πρέπει να έχουν χαρακτηριστικά τυχαιότητας, έχει αυξηθεί εκθετικά. Τυχερά παιχνίδια και κληρώσεις είναι από τα πρώτα πεδία εφαρμογής που θα μπορούσε να σκεφτεί κανείς. Αλλά δεν είναι μόνο αυτά, το τεράστιο φάσμα εφαρμογών τους εκτείνεται από τυπικές καθημερινές συνήθεις όπως μια απλή συνομιλία στο κινητό τηλέφωνο και μια συναλλαγή μέσω αυτού ή του υπολογιστή, μέχρι ζητήματα που αφορούν στη δημόσια υγεία όπως την προσπάθεια πρόβλεψης και αντιμετώπισης μιας μεταδοτικής ασθένειας. Η ολοένα και αυξανόμενη ζήτηση, λοιπόν, δημιούργησε την ανάγκη παραγωγής τους σε μεγάλες ποσότητες από μηχανισμούς οι οποίοι θα μπορούσαν να το κάνουν αυτό πολύ γρήγορα και με τρόπο τέτοιο ώστε να μην ¨πλήττει¨ την τυχαιότητα τους. Ιδεαλιστικά θα μπορούσε να φανταστεί κανείς ένα δίκαιο νόμισμα, στη μια πλευρά του οποίου αντιστοιχίζεται ο αριθμός 0 και στην άλλη ο αριθμός 1, το οποίο υποβάλλουμε σε συνεχείς ρίψεις. Τότε, κάθε ένας από του δυο αριθμούς θα είχε πιθανότητα εμφάνισης $^1/_2$ και επιπλέον οι ρίψεις θα είναι ανεξάρτητες μεταξύ τους (αφού το αποτέλεσμα οποιασδήποτε από τις προηγούμενες, δεν επηρεάζει το αποτέλεσμα οποιασδήποτε από τις επόμενες). Η δυαδική ακολουθία που θα προκύψει θα είναι τότε μια πραγματικά τυχαία δυαδική ακολουθία, από την οποία μπορεί να παραχθεί με τη σειρά της μια τυχαία ακολουθία αριθμών. Επομένως οι συνεχόμενες ρίψεις ενός δίκαιου νομίσματος θα μπορούσε να πει κανείς ότι αποτελεί την τέλεια γεννήτρια τυχαίων αριθμών. Δυστυχώς, όμως, αυτό αποτελεί μια ουτοπική διαδικασία η οποία δεν μπορεί να ανταποκριθεί στις σύγχρονες ανάγκες. Επιπλέον, όπως θα εξηγηθεί και στο Κεφάλαιο 3, γεννήτριες που βασίζονται εξ ολοκλήρου σε μια φυσική πηγή τυχαιότητας δεν προτιμούνται από την πλειονότητα των σύγχρονων εφαρμογών. Πλέον το βάρος έχει μετατοπιστεί στην κατασκευή μηχανισμών που παράγουν ακολουθίες αριθμών που ¨φαίνονται¨ να είναι τυχαίοι και οι ιδιότητες των οποίων είναι το κλειδί για την λύση των προβλημάτων που αντιμετωπίζονται. Ασχολούμαστε, δηλαδή, με γεννήτριες που παράγουν ψευδοτυχαίες ακολουθίες αριθμών. Όπως αναφέρθηκε, η χρήση τους είναι απαραίτητη σε πολλούς τομείς, η παρούσα, όμως, εργασία επικεντρώνεται σε αυτές που προορίζονται να χρησιμοποιηθούν από σύγχρονες κρυπτογραφικές εφαρμογές. Για παράδειγμα μια γεννήτρια αυτής της μορφής χρησιμοποιείται για να παράγει την κλειδοροή, βάση της οποίας θα κρυπτογραφηθεί ένα κείμενο κατά τη διάρκεια ενός κρυπτογραφικού αλγόριθμου ροής. Η παραγωγή ακολουθιών δεν είναι όμως το

μοναδικό ζητούμενο. Οι παραγόμενες ακολουθίες που χρησιμοποιούνται στην Κρυπτογραφία θα πρέπει να πληρούν χαρακτηριστικά που είναι απαραίτητα για την ασφάλεια των πληροφοριών. Κάποια από αυτά, τα χαρακτηριστικά, είναι καθολικά και κάποια ποικίλουν ανάλογα με την κρυπτογραφική εφαρμογή για την οποία προορίζονται. Για αυτό το λόγο είναι κρίσιμη η εκ των προτέρων γνώση της χρήσης της γεννήτριας, έτσι ώστε να σχεδιαστεί κατάλληλα για να μπορέσει να ανταποκριθεί στις απαιτήσεις. Ούτε όμως αυτό είναι από μόνο του αρκετό, αφού σχεδιαστεί είναι απαραίτητο, να ελεγχθεί με κάποιον τρόπο, αν όντως παράγει κατάλληλες ακολουθίες. Η δυσκολία και η σημασία του ελέγχου αυτού, είναι αντίστοιχες της κατασκευής της ίδιας της γεννήτριας. Όσο οξύμωρη και αν φαντάζει η χρήση ψευδοτυχαίων ακολουθιών, αλλά τόσο φαντάζει και η οποιαδήποτε συστηματική εξέταση μιας ακολουθίας ως προς την τυχαιότητα της. Όλα αυτά αποτελούν κομμάτι ενός ευρύτερου συμβιβασμού που πρέπει να γίνει έτσι ώστε να παραμετροποιήσουμε με κάποιο τρόπο την τύχη και να δημιουργηθούν τα εχέγγυα για να ανταποκριθούμε στις απαιτήσεις των προβλημάτων που παρουσιάζονται. Αφού τα προβλήματα αυτά αποτελούν τμήματα αλληλένδετων και πολύπλοκων διαδικασιών, οι οποίες είναι ενταγμένες σε ένα ευρύτερο Μαθηματικό πλαίσιο.

1 Μαθηματικό υπόβαθρο

Εισαγωγή

Στο κεφάλαιο αυτό θα παρουσιαστεί το βασικό Μαθηματικό υπόβαθρο , στο οποίο θα βασιστούμε για να αναπτύξουμε τη θεωρία των Γεννητριών Ψευδοτυχαίων Ακολουθιών Αριθμών (ΓΨΑ). Πιο συγκεκριμένα θα αναφερθούμε σε βασικές γνώσεις Άλγεβρας για να μπορέσουμε στη συνέχεια να εμβαθύνουμε σε ένα πιο συγκεκριμένο κλάδο της τη θεωρία Πεπερασμένων Σωμάτων.

Η θεωρία Πεπερασμένων Σωμάτων έχει κύριο ρόλο στη σχεδίαση και ανάλυση ΓΨΑ και γι αυτό το λόγο αποτελεί το βασικό αντικείμενο του παρόντος κεφαλαίου.

Τέλος , αφού θα δοθούν τα απαραίτητα εργαλεία, θα παρουσιαστούν τα συστήματα καταγραφής μετατόπισης και οι ακολουθίες καταγραφής μετατόπισης οι οποίες ουσιαστικά αποτελούν αποτέλεσμα ΓΨΑ σε καταχωρητές ολίσθησης με ανάδραση.

1.1 Αλγεβρικές Δομές

Στην παρούσα ενότητα γίνεται παρουσίαση των αλγεβρικών δομών: των ομάδων, των δακτυλίων, των σωμάτων και των πολυωνύμων.

1.1.1 Ομάδες

Ορισμός 1.1.1 Έστω F ένα μη κενό σύνολο, Μια συνάρτηση $f: F \times F \to F$ ονομάζεται (κλειστή) πράξη (πάνω στο σύνολο F).

Παραδείματα1.1.2

Η πρόσθεση $+ : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ είναι πράξη στο $\mathbb{R}$

Η αφαίρεση στο $\mathbb{N}$ $- : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ δεν είναι (κλειστή) πράξη

(5-8=-3 $\notin \mathbb{N}$)

Ορισμός 1.1.3 Έστω ένα σύνολο $F \neq \emptyset$, τότε το F μαζί με μία ή περισσότερες πράξεις (πάνω στο F), ονομάζεται αλγεβρική δομή.

Ορισμός 1.1.4 Ένα σύνολο G εφοδιασμένο με μια πράξη $\circ\ G \times G \to G$ θα ονομάζεται ομάδα, αν ικανοποιεί τις εξής ιδιότητες:

$\exists$στοιχείο $e \epsilon G$ τέτοιο ώστε: για κάθε $a \in G$ να ισχύει ότι $a \circ e = e \circ a = a$.

Για κάθε $a \in G$ $\exists a' \in G$ με την ιδιότητα $a \circ a' = e$.

Για κάθε a, b, c ισχύει η προσεταιριστική ιδιότητα $a \circ (b \circ c) = (a \circ b) \circ c$.

Επομένως το $(G, \circ)$ ονομάζεται ομάδα αν ικανοποιεί *τα παραπάνω.*

Η τάξη της ομάδας G, είναι η ισχύς του συνόλου G και συμβολίζεται με $|G|$ή $o(\mathrm{G})$.

Αν $|G| < \infty$, τότε η G λέγεται πεπερασμένη, αλλιώς λέγεται άπειρη.

Παρατηρήσεις1.1.5

i. Συνηθίζεται για την πράξη να χρησιμοποιούμε τον όρο «πολλαπλασιασμός» και για το αποτέλεσμα $a \circ b$ τον όρο «γινόμενο», μάλιστα για λόγους απλούστευσης παραλείπουμε το σύμβολο«∘»και γράφουμε ab.Πρέπει να δοθεί όμως μεγάλη προσοχή στο γεγονός ότι μερικές φορές, όταν αναφερόμαστε σε συγκεκριμένες ομάδες, για την πράξη και το αποτέλεσμά της μπορεί να χρησιμοποιούνται άλλοι όροι και συμβολισμοί, Παραδείγματος χάριν, για την προσθετική ομάδα των ακεραίων η πράξη καλείται «πρόσθεση», το αποτέλεσμα «άθροισμα» και έχουμε το συμβολισμό $a + b$.

ii. Αν μια ομάδα ικανοποιεί επιπλέον την ιδιότητα: $a \circ b = b \circ a$ *για κάθε* $a, b \in G$ τότε η ομάδα λέγεται μεταθετική ή αβελιανή

iii. Το στοιχείο e ∈ G της ιδιότητας i του ορισμού 1.1.4 είναι μοναδικό, καλείται ουδέτερο ή μοναδιαίο ή ταυτοτικό στοιχείο της ομάδας G και θα συμβολίζεται με 1_G ή πιο απλά 1. Εκτός αν σε κάποιες περιπτώσεις επιλέξουμε διαφορετικό συμβολισμό.

Παραδείγματος χάριν στην προσθετική ομάδα των ακεραίων χρησιμοποιούμε συνήθως τον συμβολισμό 0_G.

iv. Το α′∈ G της ιδιότητας ii του ορισμού 1.1.4, είναι μοναδικό, λέγεται αντίστροφο του α και συμβολίζεται με α^{-1}.Εκτός, αν πάλι, σε κάποιες περιπτώσεις επιλέξουμε διαφορετικά. Παραδείγματος χάριν στην προσθετική ομάδα των πραγματικών το συμβολίζουμε με -α.

v. Πολλές φορές χρησιμοποιούμε τον συμβολισμό (G,∘,e) ,όπου e το μοναδιαίο στοιχείο της ομάδας, αντί του (G,∘) για την ομάδα , για να αποφευχθούν συγχύσεις που μπορεί να προκληθούν. Παραδείγματος χάριν όταν χρησιμοποιούμε πολλαπλασιαστικό ή αθροιστικό συμβολισμό.

Σχόλια :Έστω $\alpha_i \in$ G,όπου $i = 1,2,\ldots.n$, Για εκφράσεις της μορφής $\alpha_1\alpha_2\ldots\alpha_n$ είναι σαφής η υπολογισιμότητά τους και δεν επηρεάζεται από τον τρόπο χρησιμοποίησης ή μη παρενθέσεων, αφού λόγω της προσεταιριστικής ιδιότητας δίνουν ως αποτέλεσμα το ίδιο πάντα στοιχείο της ομάδας G.

Επιπλέον, αν n $\in \mathbb{N}$: το στοιχείο που προκύπτει από την πράξη $\underbrace{\alpha + \alpha + \cdots + \alpha}_{n\ \tau o\ \pi\lambda\acute{\eta}\theta o\varsigma}$ χρησιμοποιώντας αθροιστικό συμβολισμό, συμβολίζεται με nα και ισχύουν οι ιδιότητες: nα+mα=(n+m)α

$$m(n\alpha)=(mn)\alpha$$

$$(-n)\alpha=n(-\alpha)$$

$$0\cdot\alpha=e=0_G,\ ^{1}$$

Ενώ το στοιχείο που προκύπτει από την πράξη $\underbrace{\alpha \cdot \alpha \cdot \ldots \cdot \alpha}_{n\ \tau o\ \pi\lambda\acute{\eta}\theta o\varsigma}$ χρησιμοποιώντας τον πολλαπλασιαστικό συμβολισμό, συμβολίζεται με α^n και ισχύουν οι ιδιότητες:

$$\alpha^n\cdot\alpha^m=\alpha^{n+m}$$

$$(\alpha^n)^m=\alpha^{nm}$$

$$\alpha^{-n}=(\alpha^{-1})^n$$

$$\alpha^0=e=1_G,\ ;\text{όπου}\ ^{1}$$

Συμβολισμός Έστω n θετικός ακέραιος, με $\mathbb{Z}_n$ συμβολίζουμε το σύνολο των υπολοίπων της διαίρεσης όλων των ακέραιων με το n, δηλαδή $\mathbb{Z}_n = \{0,1 \ldots \ldots, n-1\}$.

Το $\mathbb{Z}_n^*$ είναι το σύνολο όλων των μη μηδενικών στοιχείων του $\mathbb{Z}_n$.

Με $\alpha mod n$ συμβολίζουμε το υπόλοιπο της διαίρεσης του ακέραιου α με το n.

Για λόγους απλούστευσης θα γράφουμε στον $\mathbb{Z}_n$ "$\alpha + b$" αντί "$\alpha + b\ mod n$" και "$\alpha \cdot b$" αντί " $\alpha \cdot b mod n$", όπου + και · η συνήθης πρόσθεση και ο συνήθης πολλαπλασιασμός αντίστοιχα.

Παραδείγματα 1.1.6

i. Οι αλγεβρικές δομές : $(\mathbb{R},+,0)$,$(\mathbb{R}^*,\cdot,1)$,$(\mathbb{Z},+,0)$ αποτελούν ομάδες, όπως επίσης και οι: $(\mathbb{Z}_2,+,0)$, $(\mathbb{Z}_5,+,0)$, $(\mathbb{Z}_5^*,\cdot,1)$.
ii. Το σύνολο των φυσικών αριθμών με πράξη τη συνήθη πρόσθεση (φυσικών αριθμών) ΔΕΝ αποτελεί ομάδα.

Πρόταση 1.1.7 Έστω $\alpha,\beta \in \mathbb{Z}$ με $(\alpha,\beta) \neq (0,0)$, τότε υπάρχει ένας μέγιστος κοινός διαιρέτης των α,β , έστω δ, και μάλιστα $\delta = \alpha\chi + \beta\psi$, για κατάλληλους $\chi,\psi \in \mathbb{Z}$,συμβολικά $\delta = gcd(\alpha,\beta) = \mu\kappa\delta(\alpha,\beta)$.

---

[1] όπου στα αριστερά των ισοτήτων έχουμε τον φυσικό αριθμό 0, ενώ στα δεξιά το ουδέτερο στοιχείο της ομάδας με τον αντίστοιχο συμβολισμό .

Παρατήρηση 1,1,8 Έστω ρ πρώτος και $\alpha \in \mathbb{Z}$ με $0 < \alpha < \rho$, τότε $\exists$ χ,ψ $\in \mathbb{Z}$ τέτοια ώστε $\alpha\chi + \rho\psi = 1 = \mu\kappa\delta(\alpha, \rho)$.

Πρόταση 1.1.9 Έστω ρ πρώτος αριθμός και n θετικός ακέραιος τότε ισχύουν τα εξής:

Η $(Z_n, +, 0)$ είναι ομάδα και καλείται προσθετική ομάδα των ακεραίων $modulo\, n$.

Η $(\mathbb{Z}_\rho^*, \cdot, 1)$ είναι ομάδα και καλείται πολλαπλασιαστική ομάδα των ακεραίων $modulo\, \rho$.

Ορισμός 1.1.10 Έστω G μια ομάδα τότε:

Αν $\forall\, \psi \in G \;\; \exists\, \chi \in G$ και i $\in \mathbb{Z}$ τέτοια ώστε $\psi = \chi^i$

η ομάδα θα ονομάζεται κυκλική, το στοιχείο χ θα λέγεται γεννήτορας της κυκλικής ομάδας G και θα γράφουμε $G = < \alpha >$.

Παραδείγματα1.1.11 i. Η ομάδα $(\mathbb{Z}, +, 0)$ έχει γεννήτορες το -1 και το 1.

ii. Η ομάδα $(\mathbb{Z}_3^*, \cdot, 0)$ έχει γεννήτορα το 2.

1.1.2 Δακτύλιοι και Σώματα

Ορισμός 1.1,12 Ένα σύνολο R εφοδιασμένο με δύο πράξεις $+ : R \times R \to R$ και $\cdot : R \times R \to R$ ονομάζεται δακτύλιος αν ισχύουν οι παρακάτω ιδιότητες:

1. $(a + b) + c = a + (b + c), \forall\, a, b, c \in R$
2. $\exists$ στοιχείο 0_R τέτοιο ώστε $a + 0_R = 0_R + a = a, \forall\, a \in R$
3. $\forall\, a \in R \;\exists$ στοιχείο $a' \in R$ τέτοιο ώστε $a + a' = a' + a = 0_R$
4. $a + b = b + a, \;\forall\, a, b \in R$
5. $(a \cdot b) \cdot c = a \cdot (b \cdot c), \;\forall\, a, b, c \in R$
6. $a \cdot (b + c) = a \cdot b + a \cdot c, \forall\, a, b, c \in R$
7. $(a + b) \cdot c = a \cdot c + b \cdot c, \forall\, a, b, c \in R$

Το 0_R ονομάζεται μηδενικό στοιχείο του δακτυλίου R και το a' αντίθετο.

Παρατηρήσεις 1.1.13

1. Το μηδενικό στοιχείο του δακτυλίου R συμβολίζεται με 0_R και είναι μοναδικό.
2. $\forall\, a \in R$, υπάρχει μοναδικό αντίθετο στοιχείο $a' \in R$, συνήθως συμβολίζεται με $-a$.
3. Αν $\forall\, a, b \in R$ ισχύει $a \cdot b = b \cdot a$ τότε ο δακτύλιος R θα λέγεται μεταθετικός.

4. Αν στο δακτύλιο R $\exists$ στοιχείο 1_R τέτοιο ώστε $1_R \cdot a = a \cdot 1_R = a, \forall\, a \in R$ τότε το 1_R θα ονομάζεται μοναδιαίο στοιχείο (ή μονάδα) του R και θα λέμε ότι ο R είναι δακτύλιος με μοναδιαίο στοιχείο (ή με μονάδα).
5. Η πράξη $+$ του ορισμού 1.1.12 συνήθως καλείται πρόσθεση του δακτυλίου R.
6. Η πράξη $\cdot$ του ορισμού 1.1.12 συνήθως καλείται πολλαπλασιασμός του δακτυλίου R.
7. Η ιδιότητα 1 του ορισμού 1.1.12 συνήθως καλείται προσεταιριστική ιδιότητα της πρόσθεσης.
8. Η ιδιότητα 5 του ορισμού 1.1.12 συνήθως καλείται προσεταιριστική ιδιότητα του πολλαπλασισμού.
9. Οι ιδιότητες 6 και 7 το ορισμού 1.1.12 συνήθως καλούνται επιμεριστικοί νόμοι
10. Συνηθίζεται να χρησιμοποιούμε για λόγους απλούστευσης τους συμβολισμούς: ab αντί του $a \cdot b$, 0 αντί του 0_R και 1 αντί του 1_R (όταν αυτό υπάρχει)
11. Για να δηλώσουμε τον συμβολισμό των πράξεων $+$ και $\cdot$ σε έναν δακτύλιο R συνήθως γράφουμε $(R, +, \cdot)$.

Παραδείγματα 1.1.14 Οι αλγεβρικές δομές:

1. $(\mathbb{Z}, +, \cdot), (\mathbb{Q}, +, \cdot)$, $(\mathbb{R}, +, \cdot), (\mathbb{C}, +, \cdot)$ είναι μεταθετικοί δακτύλιοι με μονάδα, εδώ με $+$ και $\cdot$ συμβολίζονται οι συνήθεις πράξεις της πρόσθεσης και του πολλαπλασιασμού αντίστοιχα.
2. $(\mathbb{Z}_n, +, \cdot)$ είναι μεταθετικός δακτύλιος με μοναδιαίο στοιχείο, εδώ με $+$ και $\cdot$ συμβολίζονται οι πράξεις της πρόσθεσης και του πολλαπλασιασμού όπως αυτές έχουν οριστεί για τον $\mathbb{Z}_n$, ο οποίος καλείται δακτύλιος κλάσεων υπολοίπων modulo n.
3. Το σύνολο $\mathrm{M}_n(\mathbb{C})$ των $n \times n$ μιγαδικών πινάκων με τις πράξεις της πρόσθεσης και του πολλαπλασιασμού πινάκων είναι δακτύλιος με μοναδιαίο στοιχείο.

Παρατήρηση 1.1.15 Αν ένας δακτύλιος R έχει μονάδα 1_R και $1_R{=}0_R$ τότε $\forall$ r $\in R$ ισχύει ότι: r=1·r=0·r=0. Άρα $R = \{0\}$.

Ορισμοί 1.1.16 Ένας δακτύλιος $(R, +, \cdot)$ θα ονομάζεται:

1. Ακέραια περιοχή αν είναι μεταθετικός δακτύλιος με μονάδα $1_R \neq 0_R$και αν επιπλέον $\forall\, a, b \in R \setminus \{0\}$ έχουμε ότι $ab \neq 0$
2. Δακτύλιος διαίρεσης αν τα μη μηδενικά στοιχεία του R αποτελούν ομάδα ως προς την πράξη του πολλαπλασιασμού.

Παραδείγματα1.1.17

i. Οι δακτύλιοι $(\mathbb{Z},+,\cdot),(\mathbb{Q},+,\cdot)$, $(\mathbb{R},+,\cdot),(\mathbb{C},+,\cdot)$ είναι ακέραιες περιοχές.

ii. Ο δακτύλιος $D_n(\mathbb{R})$={διαγώνιοι n× n πίνακες με πραγματικούς συντελεστές}, με τις γνωστές πράξεις της πρόσθεσης και του πολλαπλασιασμού πινάκων είναι μεταθετικός δακτύλιος με μονάδα αλλά ΔΕΝ είναι ακέραια περιοχή.

iii. *Ο* $\mathbb{Z}_8$ *δεν είναι ακέραια περιοχή*

Πρόταση 1,1,18 *Ο* $\mathbb{Z}_n$ *είναι ακέραια περιοχή* $\Leftrightarrow$ *ο* n *είναι πρώτος ή μηδέν.*

Πρόταση 1,1,19 Ένας μεταθετικός δακτύλιος με μοναδιαίο στοιχείο $1_R \neq 0_R$ στον οποίο κάθε μη μηδενικό στοιχείο είναι αντιστρέψιμο λέγεται σώμα.

Παραδείγματα 1.1.20

i. Οι δακτύλιοι $(\mathbb{Q},+,\cdot)$, $(\mathbb{R},+,\cdot),(\mathbb{C},+,\cdot)$ είναι σώματα.

ii. Ο δακτύλιος $(\mathbb{Z},+,\cdot)$ είναι ακέραια περιοχή , αλλά ΔΕΝ είναι σώμα.

Παρατήρηση 1.1.21 Κάθε σώμα είναι ακέραια περιοχή, αλλά κάθε ακέραια περιοχή δεν είναι σώμα.

Πρόταση 1,1,22 Κάθε πεπερασμένη ακέραια περιοχή είναι σώμα.

Πόρισμα 1,1,23 Ο δακτύλιος $\mathbb{Z}_n$ είναι σώμα $\Leftrightarrow$ ο n *είναι πρώτος.*

Ορισμοί 1.1.24

1. Πεπερασμένο σώμα καλείται ένα σώμα του οποίου τα (διακεκριμένα) στοιχεία είναι πεπερασμένα το πλήθος.
2. Τάξη ενός πεπερασμένου σώματος ονομάζεται το πλήθος των (διακεκριμένων) στοιχείων του.
3. Θα συμβολίζουμε με $\mathbb{F}_q$ ένα πεπερασμένο σώμα με q το πλήθος στοιχεία.

(Όπως θα δούμε στην παρατήρηση 1.25 όλα τα πεπερασμένα σώματα με q το πλήθος στοιχεία είναι (ισομορφικά) ίσα, άρα είναι σαν να έχουμε (ισομορφικά) μόνο ένα πεπερασμένο σώμα με q το πλήθος στοιχεία)

Ένας δεύτερος συμβολισμός για το $\mathbb{F}_q$είναι το $GF(q)$ - προς τιμήν του Evariste Galois (1811-1832)- και συχνά αναφέρεται ως πεδίο Galois.

Παραδείγματα 1.1.25 Οι αλγεβρικές δομές:

a. $(\mathbb{Q},+,\cdot)$, $(\mathbb{R},+,\cdot),(\mathbb{C},+,\cdot)$ είναι σώματα.
b. $(\mathbb{Z}_7,+,\cdot)$ είναι πεπερασμένο σώμα.
c. $(\mathbb{Z}_2,+,\cdot)$ είναι πεπερασμένο σώμα τάξης 2. Έχει ως στοιχεία τα:0 και1 , τα οποία καλούνται δυαδικά στοιχεία.Επιπλέον οι πίνακες που αντιστοιχούν στις πράξεις της πρόσθεσης και του πολλαπλασιασμού είναι:

+	0	1
0	0	1
1	1	0

·	0	1
0	0	0
1	0	1

Σχόλιο Αν με ρ συμβολίσουμε έναν οποιονδήποτε πρώτο ακέραιο το $(\mathbb{Z}_\rho,+,\cdot)$ είναι πεπερασμένο σώμα και ονομάζεται (πεπερασμένο) σώμα κλάσεων υπολοίπων modulo ρ. Συμβολίζεται συνήθως με $\mathbb{Z}_\rho$, Όμως , όπως είπαμε και στον ορισμό1.1.24,3 και θα δούμε στην παρατήρηση 1,25, είναι σαν να έχουμε μόνο ένα πεπερασμένο σώμα (ισομορφικά) με ρ το πλήθος στοιχεία και το $\mathbb{Z}_\rho$ το συμβολίζουμε κατευθείαν με $\mathbb{F}_\rho$ή GF(ρ) και γενικότερα χρησιμοποιούνται και οι τρεις συμβολισμοί με την ίδια ευκολία.

Πόρισμα 1,1,26 Αν $n \in \mathbb{N}$ με $n \geq 2$ τα επόμενα είναι ισοδύναμα:

i. Ο n είναι πρώτος.
ii. Ο $\mathbb{Z}_n$ είναι ακέραια περιοχή.
iii. Ο $\mathbb{Z}_n$ είναι σώμα.

Ορισμός 1.1.27 Έστω R ένας δακτύλιος και S ένα μη κενό υποσύνολο του R , τέτοιο ώστε οι περιορισμοί των δύο πράξεων του $R, +: R \times R \to R$ και , $\cdot : R \times R \to R$, στο S να είναι πράξεις στο S.
Αν το S με τις πράξεις αυτές είναι δακτύλιος, θα λέμε ότι το S είναι υποδακτύλιος του R.

Παραδείγματα 1.1.28

i. Το σύνολο των άρτιων ακέραιων $2\,\mathbb{Z}$ είναι υποδακτύλιος του $\mathbb{Z}$
ii. Το σύνολο των περιττών ακέραιων ΔΕΝ είναι υποδακτύλιος του $\mathbb{Z}$
iii. Το σύνολο των άρτιων φυσικών $2\mathbb{N}$ ΔΕΝ είναι υποδακτύλιος του $\mathbb{N}$

Ορισμοί 1.1.29

i. Έστω F ένα σώμα και S ένας υποδακτύλιός του. Αν ο S είναι σώμα θα λέμε ότι είναι υπόσωμα του F. Αντίστοιχα το F θα ονομάζεται επέκταση του σώματος S.

ii. Ένα σώμα που δεν περιέχει κανένα γνήσιο υπόσωμα ονομάζεται πρωταρχικό σώμα.

Αν ρ πρώτος, τότε το πεπερασμένο σώμα $\mathbb{F}_\rho$ είναι πρωταρχικό.

Παράδειγμα1.1.30 Το $(\mathbb{Q}, +, \cdot)$ είναι υπόσωμα του $(\mathbb{C}, +, \cdot)$.

1.1.3 Πολυώνυμα

Ορισμός 1.1.31 Έστω R ένας δακτύλιος, ένα πολυώνυμο f(x) με συντελεστές στοιχεία του R είναι ένα άθροισμα της μορφής:

$$\sum_{i=0}^{\infty} \alpha_i \chi^i = \alpha_0 + \alpha_1 \chi + \cdots + \alpha_n \chi^n + \cdots$$

όπου:

$\alpha_i \in \mathbb{R}$ για κάθε $i \in \mathbb{N}$ και χ ανεξάρτητη μεταβλητή,με $\alpha_i = 0$ γενικά εκτός από ένα πεπερασμένο πλήθος i (για τα οποία $\alpha_i \neq 0$).

Τα α_i ονομάζονται συντελεστές του πολυωνύμου.

Το μεγαλύτερο $i \in \mathbb{N}$ για το οποίο $\alpha_i \neq 0$ ονομάζεται βαθμός του πολυωνύμου και συμβολίζεται με $deg(f(x)) = deg(f)$.

Το πολυώνυμο $f(\chi) = \alpha_0$ ονομάζεται μηδενικού βαθμού.

Σχόλια:

Εξ ορισμού (1.1.31) η (α_κ) είναι μια τελικά μηδενική ακολουθία του R, δηλαδή $\exists n \in \mathbb{N}$ έτσι ώστε $\alpha_i = 0$ για κάθε $i > n$,. Για αυτό το λόγο αναφερόμαστε συνήθως στο πολυώνυμο με στοιχεία του R ως: $f(\chi) = \sum_{i=0}^{n} \alpha_i \chi^i$,όπου $n = deg(f)$ και α_n ο μέγιστος όρος.

Αν ο δακτύλιος R είναι μοναδιαίος με $1_R \neq 0_R$ τότε μπορούμε να γράφουμε τον όρο $1\chi^\kappa$ ως χ^κ.

Πρόσθεση και πολλαπλασιασμός πολυωνύμων με συντελεστές στοιχεία ενός δακτυλίου R

Έστω $f(x) = \alpha_0 + \alpha_1\chi + \ldots + \alpha_n\chi^n + \cdots$.και $g(x) = b_0 + b_1\chi + \ldots + b_n\chi^n + \cdots$, με $\alpha_i, b_i \in R$ τότε :

- Για την πρόσθεση των πολυωνύμων έχουμε:

  $f(x) + g(x) = c_0 + c_1x + \cdots + c_nx^n + \ldots$, όπου $c_n = \alpha_n + b_n$ για κάθε $n \in \mathbb{N}$

- Για τον πολλαπλασιασμό των πολυωνύμων έχουμε:

  $f(x) \cdot g(x) = d_0 + d_1x + \cdots + d_nx^n + \cdots$, όπου $d_n = \sum_{i=0}^{n} \alpha_i b_{n-i}$.

Παρατήρηση 1.1.32 Το σύνολο των πολυωνύμων πάνω στο R με τις πράξεις της πρόσθεσης και του πολλαπλασιασμού , όπως αυτές ορίστηκαν παραπάνω ,αποτελεί δακτύλιο.

Ορισμός 1.1.33 Ο δακτύλιος που αποτελείται από τα πολυώνυμα πάνω στον (δακτύλιο) R, μαζί με τις ανωτέρω πράξεις της πρόσθεσης και του πολλαπλασιασμού ονομάζεται πολυωνυμικός δακτύλιος πάνω στο R και συμβολίζεται με $R[x]$.

Το μηδενικό στοιχείο του $R[x]$, είναι το πολυώνυμο για το οποίο $\alpha_i=0$ για κάθε $i \in \mathbb{N}$, ονομάζεται μηδενικό και συμβολίζεται με 0 ($0_R = O_{R[X]} = 0$).

Θεωρήματα 1.1.34

Αν ο R είναι μεταθετικός δακτύλιος, τότε και ο $R[x]$ που παράγεται από αυτόν είναι επίσης μεταθετικός.

Αν ο R έχει μοναδιαίο στοιχείο $1_R \neq 0_R$, τότε το 1_R είναι μοναδιαίο στοιχείο και για το $R[x]$.

Πρόταση 1.1.35 Αν $f(x), g(x)$ πολυώνυμα στον $F[x]$ τότε:

Το άθροισμα $f(x) + g(x)$ είναι πολυώνυμο στον $F[x]$.

Το γινόμενο $f(x) \cdot g(x)$ είναι πολυώνυμο στον $F[x]$.

Παρατήρηση 1.1.36 Αν ο δακτύλιος R είναι σώμα ,τότε ο δακτύλιος του πολυωνύμου $R[x]$, που παράγεται από τον R, είναι ακέραια περιοχή (και όχι κατ' ανάγκην σώμα).

Παρατηρήσεις 1.1.37 Για τον βαθμό δύο πολυωνύμων $f(x), g(x) \in F[x]$ ισχύουν τα ακόλουθα:

i. Τα σταθερά πολυώνυμα έχουν βαθμό μηδέν. Δηλαδή $degc = 0, c \in F \setminus \{0\}$.

ii. Βαθμός του μηδενικού πολυωνύμου δεν ορίζεται.
(Σε κάποιες βιβλιογραφίες μπορεί να δούμε $deg0 = -\infty$ ή $deg0 = -1$).

iii. $deg((f + g) \leq \max\{deg(f), deg(g)\}$.

iv. $deg(fg) = deg(f) + deg(g)$.

Ορισμός 1.1.38 Αν F σώμα και $f(x) \in F[x]$, με $deg(f) = n$ και $\alpha_n = 1$ (δηλαδή ο μεγιστοβάθμιος όρος ισούται με 1) τότε το $f(x)$ ονομάζεται μονικό πολυώνυμο.

Ορισμός 1.1.39 Αν $f(x), g(x) \in F[x]$ θα λέμε ότι τα πολυώνυμα f, g είναι ίσα αν:

- $deg(f) = deg(g) = n$.

και

- $\forall\, i = 0,1,\dots,n\ \alpha_i = \beta_i$ οι συντελεστές των f, g αντίστοιχα.

Σχόλιο Αν $F = \mathbb{R}$ ή $F = \mathbb{C}$ τότε μπορούμε να δούμε τα πολυώνυμα $f(x)$ σαν συναρτήσεις, τις λεγόμενες πολυωνυμικές συναρτήσεις, οι οποίες έχουν τη μορφή:

$f: F \to F[x]: x \to f(x)$, όπου $F = \mathbb{R}$ ή $\mathbb{C}$.

(Εδώ τα πολυώνυμα είναι ίσα αν και μόνο αν οι πολυωνυμικές συναρτήσεις είναι ίσες).

Θεώρημα 1.1.40

Αν $\alpha(x), \beta(x) \in F[x]$ με $\beta(x) \neq 0$, τότε υπάρχουν μοναδικά πολυώνυμα $\pi(x), \upsilon(x) \in F[x]$ τέτοια ώστε $\alpha(\chi) = \beta(\chi) \cdot \pi(\chi) + \upsilon(\chi)$, όπου $\upsilon(\chi) = 0$ ή $deg(\upsilon) < deg(\beta)$.

Ορισμός 1.1.41

Ένα πολυώνυμο $g(x) \in F[x]$ θα λέμε ότι διαιρεί το πολυώνυμο $f(x) \in F[x]$, αν $\exists$ π(x) ∈ F[x] τέτοιο ώστε $f(x) = g(x) \cdot \pi(\chi)$ και θα γράφουμε $g \setminus f$.

Ορισμός 1.1.42

Έστω R μια ακέραια περιοχή. Ένα πολυώνυμο $f \in R[x]$ θα λέγεται ανάγωγο:

αν $\nexists$ g(x), h(x) ∈ R[x], με $deg(g), deg(h) \geq 1$ τέτοια ώστε $f(x) = g(x) \cdot h(x)$.

Σχόλιο Ο χώρος R στον οποίο βρισκόμαστε είναι κρίσιμης σημασίας για το αν ένα πολυώνυμο είναι ανάγωγο ή όχι.

Παράδειγμα1.1.43 Το πολυώνυμο $f(x)=\chi^2 - 2$.

Αν f(x) ∈ ℚ [x] είναι ανάγωγο γιατί $\sqrt{2} \notin \mathbb{Q}$,

ενώ αν f(x) ∈ ℝ[x] δεν είναι ανάγωγο αφού $f(x)=(x-\sqrt{2})(x+\sqrt{2})$

Ορισμός 1.1.44 Έστω F ένα σώμα και $f(x), g(x) \in F[x]$. όπου $f(x) \neq 0$ ή $g(x) \neq 0$.

Εάν $\exists\, d(x) \in F[x]$ με τις ιδιότητες:

i. $d(x)$ μονικό

ii. $.d(x) \setminus f(x)$ και $d(x) \setminus g(x)$ στο $F[x]$

iii. Αν $c(x) \in F[x]$, $c(x) \setminus f(x)$ και $c(x) \setminus g(x)$ τότε $c(x) \setminus d(x)$

Τότε το $d(x)$ ονομάζεται μέγιστος κοινός διαιρέτης των πολυωνύμων $f(x), g(x)$ (στο $F[x]$).

Παρατηρήσεις 1.1.45

a. Από την ιδιότητα ii.έπεται ότι κάθε άλλος κοινός διαιρέτης των f, g έχει βαθμό $\leq deg(d)$.
b. Αν $\mu\kappa\delta(f(x), g(x)) = 1$ τότε τα των f, g ονομάζονται πρώτα μεταξύ τους ή σχετικά πρώτα.

Θεώρημα 1.1.46 Έστω F σώμα $f(x), g(x) \in F[x]$ με $f(x) \neq 0$ ή $g(x) \neq 0$
τότε ∃ μοναδικό $d(x) = \mu\kappa\delta(f(x), g(x))$
και επιπλέον $\exists\, \alpha(x), f(x) \in F[x] : d(x) = \alpha(x)f(x) + \beta(\chi)g(x)$.

Ορισμός 1.1.47 Το $b \in F$ ονομάζεται ρίζα του πολυωνύμου $f(x) \in F[x]$ αν $f(b) = 0$.

Θεώρημα 1.1.48 (Ανάλυση του πολυωνύμου $f(x)$ σε γινόμενο μονικών ανάγωγων πολυωνύμων)

Έστω F σώμα, κάθε πολυώνυμο $f(x) \in F[x]$, με $deg(f) > 0$ γράφεται κατά μοναδικό τρόπο (με εξαίρεση την διάταξη των παραγόντων) ως:

$$f(\chi) = c \cdot \rho_1(\chi)^{r_1} \cdot \ldots \cdot \rho_m(\chi)^{r_m} \quad ,$$

όπου $c \in F \setminus \{0\}$, τα $\rho_i(\chi) \in F[\chi], i = 1,2,\ldots,m$, είναι διακεκριμένα μονικά ανάγωγα πολυώνυμα και οι r_i θετικοί ακέραιοι, $i = 1,2,\ldots,m$.

Παράδειγμα1.1.49 Αν $f(x)=\chi^5-\chi^4+2\chi^3-2\chi^2+\chi-1 \in \mathbb{R}[x]$, τότε η ανάλυσή του σε γινόμενο μονικών ανάγωγων πολυωνύμων είναι $f(x)=(\chi^2+1)^2(\chi-1)$.

Ενώ αν $f(x) \in \mathbb{C}[x]$ τότε η ανάλυση είναι $f(x)=(\chi-i)^2(\chi+i)^2(\chi-1)$

Σχόλιο Κάθε πολυώνυμο $R[x]$ παραγοντοποιείται κατά μοναδικό τρόπο ως γινόμενο μονικών ανάγωγων πολυωνύμων. Αυτό δίνει στα ανάγωγα πολυώνυμα κύριο ρόλο στη δομή ενός δακτύλίου $R[x]$ και επακόλουθα και στη δομή ακολουθιών που παράγονται από καταχωρητές ολίσθησης γραμμικής ανάδρασης.

1.2 Βασική θεωρία πεπερασμένων σωμάτων

Στην παρούσα ενότητα παρουσιάζονται βασικά γνωρίσματα της θεωρίας πεπερασμένων σωμάτων όπως: η χαρακτηριστική, οι δομές, οι γεννήτορες και οι αναπαραστάσεις των στοιχείων τους.

1.2.1 Χαρακτηριστικά πεπερασμένου σώματος

Ορισμός 1.2.1 Έστω F πεπερασμένο σώμα και έστω ο ελάχιστος θετικός ακέραιος n με την ιδιότητα $n\alpha = 0, \forall\, \alpha \in F$
τότε ο n θα ονομάζεται χαρακτηριστική του σώματος F και θα λέμε ότι το σώμα F έχει χαρακτηριστική n.

Θεώρημα 1.2.2 Ένα πεπερασμένο σώμα F έχει χαρακτηριστική πρώτο αριθμό

Σχόλιο Ένα πρωταρχικό σώμα $\mathbb{F}_{\rho,}$, όπου ρ πρώτος, έχει χαρακτηριστική ρ.

1.2.2 Δομές πεπερασμένου σώματος

Θεώρημα 1.2.3 Έστω F πεπερασμένο σώμα με $|F| = q$ στοιχεία και χαρακτηριστική ρ, όπου ρ πρώτος, διακρίνουμε τις περιπτώσεις:

- $q = \rho$ τότε $F = \mathbb{F}_\rho$
- ενώ αν $q > \rho$, δηλαδή αν $q = \rho^n$ για κάποιο $n \in \mathbb{N}$ με $n \geq 1$, τότε το σώμα F είναι n-διάστατος διανυσματικός χώρος.

Ορισμός 1.2.4 Έστω F, G πεπερασμένα σώματα. Αν υπάρχει μια 1-1 και επί απεικόνιση από το F στο G, η οποία διατηρεί τις πράξεις της πρόσθεσης και του πολλαπλασιασμού, τότε τα F και G ονομάζονται ισόμορφα.

Παρατήρηση 1.2.5 Όλα τα πεπερασμένα σώματα τάξης ρ^n είναι ισόμορφα μεταξύ τους.

Ορισμός 1.2.6 Ορίζουμε ως $\mathbb{F}_{\rho^n}$ την επέκταση που λαμβάνουμε όταν στο $\mathbb{F}_\rho$ προσαρτήσουμε μια από τις ρίζες ενός ανάγωγου πολυώνυμου βαθμού n σε αυτόν (στον $\mathbb{F}_\rho$).

Σχόλιο Από τα θεωρήματα 1.2.2, 1.2.3 και την παρατήρηση 1.2.5 συμπεραίνουμε ότι ένα οποιοδήποτε πεπερασμένο σώμα θα έχει χαρακτηριστική ίση με κάποιον πρώτο ρ. Τότε ανάλογα με το $|F|$ θα κατατάσσεται αναγκαστικά σε έναν από τους δύο τύπους πεπερασμένων σωμάτων:

- είτε στο πρωταρχικό σώμα $\mathbb{F}_\rho$,
- είτε στην επέκταση $\mathbb{F}_{\rho^n}$.

Συμβολισμός Έστω F πεπερασμένο σώμα, θα συμβολίζουμε με F^* την πολλαπλασιαστική ομάδα των μη μηδενικών στοιχείων του F.

Ορισμός 1.2.7 Έστω $\alpha \in \mathbb{F}_q^*$ τότε ο μικρότερος θετικός ακέραιος για τον οποίο θα ισχύει $\alpha^n = 1$, θα ονομάζεται τάξη του α και θα το συμβολίζουμε με $ord(\alpha) = r$.

Θεώρημα 1.2.8 Έστω F πεπερασμένο σώμα, τότε η πολλαπλασιαστική ομάδα F^* είναι κυκλική.

Ορισμός 1.2..9

i. Ονομάζουμε πρωταρχικό στοιχείο ή πρωταρχική ρίζα του $\mathbb{F}_{\rho^n}$, τον γεννήτορα της κυκλικής ομάδας $F_{\rho^n}^*$
ii. Ονομάζουμε πρωταρχικό πολυώνυμο, ένα μονικό πολυώνυμο του οποίου όλες οι ρίζες είναι πρωταρχικά στοιχεία.

Παράδειγμα1.2.10 Για το $\mathbb{F}_7$ έχουμε $\mathbb{F}_7^*$={1,2,3,4,5,6}

Οι δυνάμεις του 2 είναι {2, 2^2=4, 2^3=8=1} και επομένως το 2 έχει τάξη 3 και δεν μπορεί να παράγει ολόκληρη την πολλαπλασιαστική ομάδα.

Ενώ οι δυνάμεις του 3 είναι {3, 3^2=9=2, 3^3=27=6, 3^4=81=4, 3^5=243=5, 3^6=729=1}.

Άρα το 3 έχει τάξη 6 και παράγει όλη την ομάδα, άρα είναι γεννήτορας της $\mathbb{F}_7^*$ και πρωταρχικό στοιχείο της $\mathbb{F}_7$

Παρατηρήσεις 1.2.11

i. Έστω F ένα πεπερασμένο σώμα τάξης ρ^n και $\alpha \in F$ τότε (από ον ορισμό 1.2.9) το α είναι πρωταρχικό στοιχείο του F αν η τάξη του είναι ρ^n-1.

ii. Αν ένα πολυώνυμο είναι ανάγωγο δεν έπεται ότι είναι και πρωταρχικό.

iii. Έστω $u, v \in F^*$ με $ord(u) = r$ και $ord(v) = s$ τότε ισχύουν τα ακόλουθα:

α. $ord(u^m) = r/\mu\kappa\delta(r.m)$, και προφανώς , αν επιπλέον $\mu\kappa\delta\,(r, m) = 1$ τότε $ord(u^m) = ord(u)$

β. Αν $\mu\kappa\delta(r, s) = 1$ τότε $ord(u, v) = rs$.

Παράδειγμα1.2.12 Έστω ο $\mathbb{F}_{16} = \mathbb{F}_{2^4}$,το $f(x)=x^4+x^3+x^2+1 \in \mathbb{F}_{2^4}[x]$ είναι ανάγωγο αλλά ΔΕΝ είναι πρωταρχικό.

Λήμμα1.2.13 Έστω q πρωταρχικό στοιχείο του $\mathbb{F}_q$, τότε

Το q^s είναι (και αυτό) πρωταρχικό στοιχείο του $\mathbb{F}_q \Leftrightarrow \mu\kappa\delta(s, q - 1) = 1$.

Πόρισμα 1.2.14 Κάθε πεπερασμένο σώμα περιέχει πρωταρχικά στοιχεία.

Θεώρημα 1.2.15 (Θεώρημα Fermat)

Κάθε στοιχείο α του πεπερασμένου σώματος F τάξης ρ^n είναι ρίζα της εξίσωσης

$$\chi^{\rho^n} = x$$

Άρα $\chi^{\rho^n} - \chi = \sum_{\alpha \in F}(\chi - \alpha)$

Πόρισμα 1.2.16 Έστω F σώμα με χαρακτηριστική ρ τότε $(a + b)^{\rho^n} = a^{\rho^n} + b^{\rho^n}, \forall\, n \geq 1$

1.2.3 Αναπαράσταση στοιχείων

Θεώρημα 1.2.17 Για κάθε πεπερασμένο σώμα K και κάθε πεπερασμένη επέκταση F του K, υπάρχει κανονική βάση του F πάνω στο K.

Σχόλιο Επομένως για κάθε πεπερασμένο σώμα $\mathbb{F}_{\rho^n}$, το οποίο αποτελεί επέκταση του σώματος $\mathbb{F}_\rho$, υπάρχει τουλάχιστον μία κανονική βάση (του $\mathbb{F}_{\rho^n}$) πάνω στο $\mathbb{F}_\rho$.

Υπενθυμίζουμε ότι από το θεώρημα 1.2.3 ένα πεπερασμένο σώμα $\mathbb{F}_{\rho^n}$ αποτελεί n-διάστατο διανυσματικό χώρο πάνω στο $\mathbb{F}_\rho$. Επομένως είναι κρίσιμο να μπορούμε να βρούμε μια βάση του, να μπορούμε να βρούμε -δηλαδή- ένα σύνολο n στοιχείων που να είναι γραμμικά ανεξάρτητα.

Θα παραθέσουμε λοιπόν δύο βάσεις του $\mathbb{F}_{\rho^n}$ ιδιαίτερης σημασίας:

● Θεωρούμε την πολυωνυμική βάση $\{1, \alpha, \dots, \alpha^{n-1}\}$,την οποία χρησιμοποιούμε για να παράξουμε το $\mathbb{F}_{\rho^n}$ από ένα ανάγωγο πολύώνυμο $f(x) \in \mathbb{F}[x]$, με $deg(f) = n$ και $f(\alpha) = 0$ για κάποιο $\alpha \in \mathbb{F}_{\rho^n}$.

Τότε κάθε στοιχείο του $\mathbb{F}_{\rho^n}$ μπορεί να αναπαρασταθεί από ένα πολυώνυμο του α πάνω στο $\mathbb{F}_\rho$ βαθμού <n.

● Όταν τα $\alpha, \alpha^\rho, \dots, \alpha^{\rho^{n-1}}$ είναι γραμμικά ανεξάρτητα θεωρούμε την κανονική βάση $\{\alpha, \alpha^\rho, \dots, \alpha^{\rho^{n-1}}\}$

Σε αυτήν την περίπτωση, λοιπόν, θα μας ενδιαφέρει να ελέγξουμε αν είναι γραμμικά ανεξάρτητα.

Ένα κριτήριο απόρριψης των $\alpha, \alpha^\rho, \dots, \alpha^{\rho^{n-1}}$ θα είναι η ισχύς της ακόλουθης συνθήκης : $\alpha + \alpha^\rho + \cdots + \alpha^{\rho^{n-1}} = 0$

και μόνο όταν δεν ισχύει η παραπάνω θα έχει νόημα να τα εξετάσουμε ως προς τη γραμμική ανεξαρτησία τους (η οποία πάλι δεν είναι εξασφαλισμένη).

Παράδειγμα1.2.18 Έστω ότι θέλουμε να αναπαραστήσουμε τα στοιχεία του $\mathbb{F}_9$.

Επειδή $\mathbb{F}_9 = \mathbb{F}_{3^2}$ θα τον θεωρήσουμε ως μια επέκταση βαθμού 2 του $\mathbb{F}_3$.

Έστω το ανάγωγο $f(x)=\chi^2+1 \in \mathbb{F}_3[\chi]$ και $\alpha \in \mathbb{F}_{3^2}$ τέτοιο ώστε $f(\alpha) = \alpha^2 + 1 = 0$

και τα εννέα στοιχεία του $\mathbb{F}_9$ θα δίνονται στη μορφή

$$a_0 + a_1\alpha, \text{ όπου } a_0, a_1 \in \mathbb{F}_3$$

Δηλαδή $\mathbb{F}_9 = \{0, 1, 2, \alpha, 1+\alpha, 2+\alpha, 2\alpha, 1+2\alpha, 2+2\alpha\}$

Συμπέρασμα

Από το θεώρημα 1.2.8 η πολλαπλασιαστική ομάδα $F^*_{\rho^n}$ είναι κυκλική.

Έστω τώρα ένα πρωταρχικό στοιχείο $\alpha \in \mathbb{F}_{\rho^n}$ και $\{\alpha^0, \dots, \alpha^{n-1}\}$ μία βάση του $\mathbb{F}_{\rho^n}$ στο $\mathbb{F}_\rho$.

Τότε μπορούμε να εκφράσουμε το $\mathbb{F}_{\rho^n}$ με βάση τη διανυσματική και εκθετική αναπαράσταση αντίστοιχα ως:

$\mathbb{F}_{\rho^n} = \{a_0\alpha^0 + a_1\alpha + \dots + a_{n-1}\alpha^{n-1} : a_i \in \mathbb{F}_\rho, i = 0,1,\dots,n-1\} = \{\alpha^i : 0 \le i \le \rho^n - 2 \text{ ή } i = \infty\}$ με τη συμφωνία ότι $\alpha^\infty = 0$

Παράδειγμα1.2.19 Αναπαράσταση του $\mathbb{F}_8 = \mathbb{F}_{2^3}$, όπως ορίζεται από το $f(x)=x^3+x+1 \in \mathbb{F}_2[\chi]$ και μία ρίζα του $\alpha \in \mathbb{F}_8$

Πολυωνυμική βάση			Κανονική βάση			Εκθετική
α^0	α^1	α^2	α^3	α^6	α^5	α^i
0	0	0	0	0	0	∞
1	0	0	1	1	1	0
0	1	0	0	1	1	1
0	0	1	1	0	1	2
1	1	0	1	0	0	3
0	1	1	1	1	0	4
1	1	1	0	0	1	5
1	0	1	0	1	0	6

1.3 Κατασκευές του $\mathbb{F}_{\rho^n}$

Στην παρούσα ενότητα παρουσιάζεται η μέθοδος κατασκευής ενός πεπερασμένου σώματος $\mathbb{F}_{\rho^n}$, όπου ρ πρώτος αριθμός και n θετικός ακέραιος.

Έστω n θετικός ακέραιος και ρ πρώτος.

Γνωρίζουμε από τα προηγούμενα ότι το $\mathbb{F}_\rho = \{0,1,\dots,\rho-1\}$, είναι ένα πεπερασμένο σώμα τάξης ρ, που λέγεται πρωταρχικό σώμα $\mathbb{F}_\rho$.

Βρίσκουμε ένα ανάγωγο πολυώνυμο $f(x) \in \mathbb{F}_\rho[\chi]$, με $deg(f) = n$ και θεωρούμε μία ρίζα του α,δηλαδή $f(\alpha) = 0$.

Ορίζουμε τότε

$$\mathbb{F}_{\rho^n} = \{a_0 + a_1\alpha + \cdots + a_{n-1}\alpha^{n-1} / a_i : \in \mathbb{F}_\rho, i = 0,1,\dots,n-1\} \qquad (\mathbf{1.3.1})$$

Πρέπει να ορίσουμε τις πράξεις της πρόσθεσης $(+)$ και του πολλαπλασιασμού $(\cdot)$ στο $\mathbb{F}_{\rho^n}$.

Έστω τα στοιχεία $u, v \in \mathbb{F}_{\rho^n}$, τότε σύμφωνα με την 1.3.1, αυτά θα έχουν τη μορφή:

$$u(\alpha) = \sum_{i=0}^{n-1} a_i\alpha^i \ \kappa\alpha\iota\ v(\alpha) = \sum_{i=0}^{n-1} b_i\alpha^i$$

- <u>*Ορίζουμε την πράξη της πρόσθεσης στο $\mathbb{F}_{\rho^n}$ ως*</u> :

$$u(\alpha) + v(\alpha) = \left(\sum_{i=0}^{n-1} a_i\alpha^i\right) + \left(\sum_{i=0}^{n-1} b_i\alpha^i\right) = \sum_{i=0}^{n-1}(a_i + b_i)\alpha^i$$

- Για την πράξη του πολλαπλασιασμού θα ακολουθήσουμε μια πιο περίπλοκη διαδικασία:

Θα πολλαπλασιάσουμε τα $u(\alpha), v(\alpha)$ αρχικά σύμφωνα με τον πολλαπλασιασμό πολυωνύμων

$$u(\alpha)\cdot v(\alpha) = \left(\sum_{i=0}^{n-1} a_i\alpha^i\right)\cdot\left(\sum_{i=0}^{n-1} b_i\alpha^i\right) = \sum_{k=0}^{2n-2} c_k\alpha^k = c(\alpha), \text{ όπου } c_k = \sum_{i+j=k} a_i b_j$$

και ύστερα θα διαιρέσουμε το $c(\alpha)$ με το $f(\alpha)$ και τότε θα προκύψουν δύο μοναδικά στοιχεία $w(\alpha)$ και $r(\alpha)$ τέτοια ώστε:

$c(\alpha) = w(\alpha)f(\alpha) + r(\alpha)$, όπου $deg(r) < n$

Όμως $f(\alpha) = 0$ άρα $c(\alpha) = r(\alpha)$

Ορίζουμε τότε και την πράξη του πολλαπλασιασμού στο $\mathbb{F}_{\rho^n}$ ως:

$$u(\alpha) \cdot v(\alpha) = r(\alpha) \in \mathbb{F}_{\rho^n}$$

Θεώρημα 1.3.2 Το σύνολο $\mathbb{F}_{\rho^n}$ με τις πράξεις της πρόσθεσης και του πολλαπλασιασμού που ορίσαμε πάρα πάνω αποτελεί πεπερασμένο σώμα τάξης ρ^n.

Σχόλιο Λόγω του τρόπου κατασκευής του $\mathbb{F}_{\rho^n}$, που παρατέθηκε , συνήθως λέμε ότι το $\mathbb{F}_{\rho^n}$ είναι πεπερασμένη επέκταση του $\mathbb{F}_\rho$ ή ότι το $\mathbb{F}_{\rho^n}$ παράγεται από το πρωταρχικό σώμα $\mathbb{F}_\rho$, από την προσάρτηση στο $\mathbb{F}_\rho$ μιας ρίζας του ανάγωγου πολυωνύμου f(x).

Ορισμός 1.3.3

i. Το $f(x)$ ονομάζεται ορίζον πολυώνυμο του $\mathbb{F}_{\rho^n}$ στο $\mathbb{F}_\rho$.
ii. Το α ονομάζεται ορίζον στοιχείο του $\mathbb{F}_{\rho^n}$ στο $\mathbb{F}_\rho$ και επειδή $f(\alpha) = 0$ το α αναφέρεται και ως ρίζα του $f(x)$ στο $\mathbb{F}_{\rho^n}$.

Παράδειγμα1.3.4 Ας δούμε πάλι το παράδειγμα 1.2.19

Σύμφωνα με όσα είδαμε για $\rho = 2$ και $n = 3$ έχουμε:

Έστω το ανάγωγο πολυώνυμο $f(x)=x^3+x+1 \in \mathbb{F}_2[\chi]$ και α μία ρίζα του, δηλαδή $f(\alpha)=0$ τότε το πεπερασμένο σώμα $\mathbb{F}_{2^3}$ ορίζεται ως

$$\mathbb{F}_{2^3} = \{a_0\alpha^0 + a_1\alpha + a_2\alpha^2 : a_i \in \mathbb{F}_2, i = 0,1,2\}$$

Επομένως τα στοιχεία του $\mathbb{F}_{2^3}$ που ορίζεται από το f(x) με $f(\alpha)=0$ είναι:

- ως διάνυσμα: 000 , 100 , 010 , 001 , 110 , 011 , 111 , 101
- ως πολυώνυμο, : 0 , 1 , α , α^2 , $1+\alpha$, $\alpha+\alpha^2$, $1+\alpha+\alpha^2$, $1+\alpha^2$
- ως δυνάμεις: 0 , 1 , α , α^2 , α^3 , α^4 , α^5 , α^6

Επιπλέον βλέπουμε ότι τα μη μηδενικά στοιχεία του $\mathbb{F}_{2^3}$ είναι κυκλική ομάδα τάξης 7 με γεννήτορα το α , δηλαδή το $\mathbb{F}^*_{2^3} = <\alpha>$, με $\alpha^7=1$.

Παράδειγμα1.3.5 Για n=4 και ρ=2 θέλουμε να κατασκευάσουμε το $\mathbb{F}_{2^4}$

Επιλέγουμε το ανάγωγο πολυώνυμο $f(x)=x^4+x+1 \in \mathbb{F}_2[\chi]$ και μία ρίζα α του f(x), δηλαδή $f(\alpha)=0$.

Θεωρούμε τότε το πεπερασμένο σώμα $\mathbb{F}_{2^4}$ ως κάτωθι:

$\mathbb{F}_{2^4}=\{ \alpha_0+\alpha_1\alpha+\alpha_2\alpha^{2+}\alpha_3\alpha^3/\alpha_1,\alpha_2,\alpha_3 \in \mathbb{F}_2\}$

Τότε τα στοιχεία του $\mathbb{F}_{2^4}$ που ορίστηκε με τη βοήθεια του $f(x)=x^4+x+1 \in \mathbb{F}_2[\chi]$, με $f(\alpha)=0$ είναι τα :

Ως διάνυσμα	Ως πολυώνυμο	Ως δυνάμεις
0000	0	0
1000	1	1
0100	α	α
0010	α^2	α^2
0001	α^3	α^3
1100	$\alpha+1$	α^4
0110	$\alpha^2+\alpha$	α^5
0011	$\alpha^3+\alpha^2$	α^6
1100	$\alpha^3+\alpha+1$	α^7
1010	α^2+1	α^8
0101	$\alpha^3+\alpha$	α^9
1110	$\alpha^2+\alpha+1$	α^{10}
0111	$\alpha^3+\alpha^2+\alpha$	α^{11}
1111	$\alpha^3+\alpha^2+\alpha+1$	α^{12}
1011	$\alpha^3+\alpha^2+1$	α^{13}
1001	α^3+1	α^{14}

Επιπλέον η $\mathbb{F}^*_{2^4}=<\alpha>$, είναι κυκλική ομάδα τάξης 15 με γεννήτορα το στοιχείο α , όπου $\alpha^{15}=1$

Επίσης έχουμε ότι $\alpha^4=\alpha+1$ και $\alpha^9=\alpha^3+\alpha$

Για την πρόσθεση και τον πολλαπλασιασμό στο $\mathbb{F}_{2^4}$ έχουμε:

$(\alpha^2+\alpha) \in \mathbb{F}_{2^4}$ και $(\alpha^2+1) \in \mathbb{F}_{2^4}$ τότε $(\alpha^2+\alpha)+(\alpha^2+1)=\alpha+1$

$(\alpha^3+\alpha) \in \mathbb{F}_{2^4}$ και $(\alpha^2+1) \in \mathbb{F}_{2^4}$ τότε $(\alpha^3+\alpha)\cdot(\alpha+1)=\alpha^4\alpha^9=\alpha^{4+9}=\alpha^{13}=\alpha^3+\alpha^2+1$

Σχόλιο Για την πράξη της πρόσθεσης μας εξυπηρετεί η πολυωνυμική αναπαράσταση, ενώ γι αυτήν του πολλαπλασιασμού η εκθετική.

Ορισμός 1.3.6 Έστω $f(\chi) = \alpha_0 + \alpha_1 x + \cdots + \alpha_{n-1}\chi^{n-1} + \alpha_n x^n \in \mathbb{F}_\rho[\chi]$, καλούμε αντίστροφο πολυώνυμο του $f(\chi)$ το πολυώνυμο που προκύπτει αν αντιστρέψουμε τους συντελεστές του, δηλαδή το

$$f^*(\chi) = \alpha_n + \alpha_{n-1}\chi + \cdots + \alpha_1\chi^{n-1} + \alpha_0\chi^n \in \mathbb{F}_\rho[\chi]$$

Οι ρίζες του $f^*(\chi)$ λαμβάνονται αν αντιστρέψουμε τις ρίζες του f(x) στην επέκταση $\mathbb{F}_{\rho^n}$ του $\mathbb{F}_\rho$.

Αν f(χ)πρωταρχικό πολυώνυμο τότε και $f^*(\chi)$ πρωταρχικό πολυώνυμο.

1.4 Ελάχιστα Πολυώνυμα

Έστω α ένα στοιχείο ενός πεπερασμένου σώματος $\mathbb{F}_q$. Υπενθυμίζουμε ότι από θεώρημα 1.2.3 υπάρχει ρ πρώτος και θετικός ακέραιος n τέτοιος ώστε $\mathbb{F}_q=\mathbb{F}_{\rho^n}$ και $q = \rho^n$.

Έστω επιπλέον το μονικό πολυώνυμο $f(\chi) = \chi^q - \chi = \chi^{\rho^n} - x \in \mathbb{F}_\rho[\chi]$ τότε από το θεώρημα 1.2.15 συμπεραίνουμε ότι το α είναι ρίζα του f(x), όπως και της εξίσωσης $\chi^q - \chi = 0$, με συντελεστές στο $\mathbb{F}_\rho$.

Ορισμός 1.4.1 Έστω K, F σώματα , όπου $K \subseteq F$ και $m(x) \in K[\chi]$, όπου $m(\chi)$ μονικό πολυώνυμο με $m(\alpha)=0$. Τότε το $m(\chi)$ ονομάζεται ελάχιστο πολυώνυμο του α, αν δεν υπάρχει άλλο μη μηδενικό πολυώνυμο στο $K[\chi]$ μικρότερου βαθμού (από το $m(\chi)$) που έχει ρίζα το α. Το συμβολίζουμε συνήθως με $m(\chi)$.

Σχόλιο Ιδιαίτερο ενδιαφέρον έχουν οι περιπτώσεις $K = \mathbb{F}_\rho$ και $F=\mathbb{F}_{\rho^n}$ με τις οποίες και θα ασχοληθούμε.

Παράδειγμα1.4.2 Για τον $\mathbb{F}_{16} = \mathbb{F}_{2^4}$.

Κατασκευάζουμε τον $\mathbb{F}_{2^4}$ ως επέκταση του $\mathbb{F}_2$,δείτε παράδειγμα 1.3.5 , για $f(x)=x^4+x+1 \in \mathbb{F}_2[\chi]$ και $\alpha \in \mathbb{F}_{2^4}$ με $f(\alpha)=0$ τότε:

Το ελάχιστο πολυώνυμο του: $0 \in \mathbb{F}_{16}$ είναι το $m(\chi) = \chi \in \mathbb{F}_2[\chi]$

$1 \in \mathbb{F}_{16}$ είναι το $m(\chi) = \chi + 1 \in \mathbb{F}_2[\chi]$

$\alpha \in \mathbb{F}_{16}$ είναι το $m(\chi) = \chi^4 + \chi + 1 \in \mathbb{F}_2[\chi]$

$\alpha^3 \in \mathbb{F}_{16}$ είναι το $m(\chi) = \chi^4 + \chi^3 + \chi^2 + \chi + 1 \in \mathbb{F}_2[\chi]$

$\alpha^5 \in \mathbb{F}_{16}$ είναι το $m(\chi) = \chi^2 + \chi + 1 \in \mathbb{F}_2[\chi]$

$\alpha^7 \in \mathbb{F}_{16}$ είναι το $m(\chi) = \chi^4 + \chi^3 + 1 \in \mathbb{F}_2[\chi]$

Ιδιότητες1.4.3 Έστω το $m(\chi) \in \mathbb{F}_\rho[\chi]$ το ελάχιστο πολυώνυμο του $\alpha \in \mathbb{F}_{\rho^n}$ τότε ισχύουν τα ακόλουθα:

i. Το $m(\chi)$ είναι ανάγωγο στο $\mathbb{F}_\rho[\chi]$.

ii. $m(\chi) / g(\chi)$, αν $g(\chi) \in \mathbb{F}_\rho[\chi]$ και $g(\alpha) = 0$.

iii. Αν $f(\chi) = \chi^{\rho^n} - \chi \in \mathbb{F}_\rho[\chi]$ τότε $m(\chi) / f(\chi)$.

iv. . $\deg(m) \leq n$.

v. Τα στοιχεία $\alpha, \alpha^\rho \in \mathbb{F}_{\rho^n}$ έχουν το ίδιο ελάχιστο πολυώνυμο.

vi. Αν α πρωταρχικό στοιχείο του $\mathbb{F}_{\rho^n}$ τότε $\deg(m) = n$.

Ορισμοί 1.4.4

i. Έστω $\alpha \in \mathbb{F}_{\rho^n}$ τότε ως συζυγή στοιχεία του α στο $\mathbb{F}_\rho$ ορίζουμε τα $\alpha, \alpha^{\rho} \ldots\ldots, \alpha^{\rho^{n-1}}$.

ii. Έστω $s \in \mathbb{N}$ και m_s ο ελάχιστος θετικός ακέραιος έτσι ώστε $s\rho^{m_s} = s(\mathrm{mod}\rho^n - 1)$ τότε ονομάζουμε κυκλοτομική κλάση του s(modρ) το σύνολο $C_s = \{s, s\rho, \ldots, s\rho^{m_s-1}\}$.

iii. Επικεφαλής κλάσης $\mathrm{mod}\rho^n - 1$ καλείται ο ελάχιστος ακέραιος της κυκλοτομικής κλάσης C_s, ο οποίος συμβολίζεται συνήθως με s. Το σύνολο των επικεφαλείς κλάσεων $\mathrm{mod}\rho^n - 1$, το συμβολίζουμε συνήθως με I.

Παρατηρήσεις 1.4.5

i. Όλα τα συζυγή στοιχεία του $\alpha \in \mathbb{F}_{\rho^n}$ έχουν το ίδιο ελάχιστο πολυώνυμο.

ii. Το γινόμενο των (συζυγών) εκθετών του α με το ρ διαιρεί τους ακέραιους $\mathrm{mod}\rho^n - 1$.

iii. Στα πεπερασμένα σώματα $\mathbb{F}_{2^n}$ το πλήθος των κυκλοτομικών κλάσεων ισούται με

$$|I| = \frac{1}{n}\sum_{\frac{d}{n}} \varphi\left(\frac{n}{d}\right) 2^d - 1$$

Όπου $\varphi(\cdot)$ η συνάρτηση Euler η οποία ορίζεται ως $\varphi(n) =$ πλήθος των θετικών ακεραίων i, όπου $i \leq n$ και $\mu\kappa\delta(i, n) = 1$.

Παράδειγμα1.4.6 Για ρ=2 και n=4, $\rho^n - 1 = 2^4 - 1 = 15$ και οι κυκλοτομικές κλάσεις mod15 είναι: $C_0 = \{0\}$ $C_1 = \{1,2,4,8\},\}$ $C_3 = \{3,6,9,12\}$, $C_5 = \{5,10\}$, $C_7 = \{7,11,14\}$.

● Όπως μπορεί να αντιληφθεί εύκολα ο αναγνώστης η εύρεση ελαχίστων πολυωνύμων φαίνεται χρονοβόρα και επίπονη διαδικασία αν δεν υπάρχει κάποια συγκεκριμένη μέθοδος.

Θα παραθέσουμε για αυτό *τον αλγόριθμο εύρεσης ελαχίστων πολυωνύμων:*

Είσοδος: ένα πρωταρχικό πολυώνυμο $f(\chi) \in \mathbb{F}_\rho[\chi]$ βαθμού n

βήμα 1: Κατασκεύασε το πεπερασμένο σώμα $\mathbb{F}_{\rho^n}$ βάσει της εξίσωσης $f(0) = 0$

βήμα 2: Υπολόγισε τις κυκλοτομικές κλάσεις $C_1, \ldots, C_n \ mod\ \rho^n - 1$ και το σύνολο I των επικεφαλείς κλάσεων $mod\ \rho^n - 1$

βήμα 3: $\forall\, C \in I$ υπολόγισε το πολυώνυμο $m_s(\chi) = \prod_{i \in C_s}(\chi - \alpha^i)$

Έξοδος: Όλα τα ανάγωγα πολυώνυμα στο $\mathbb{F}_\rho$ με $\deg(f)/n$

Σχόλια-Παρατηρήσεις-Ορισμοί 1.4.7

i. Όταν ο i διατρέχει τα στοιχεία μιας κυκλοτομικής κλάσης, τότε όλα τα α^i έχουν το ίδιο ελάχιστο πολυώνυμο.

ii. Το $m_s(\chi)$ είναι το ελάχιστο πολυώνυμο του α^s και των συζυγών στοιχείων του.

iii. Αν $\mu\kappa\delta(s, \rho^n - 1) = 1$ τότε από παρατήρηση 1.2.11 iii $ord(\alpha^s) = ord(\alpha) = \rho^n - 1$ και το α^s αποτελεί πρωταρχικό στοιχείο του $\mathbb{F}_{\rho^n}$. Άρα το $m_s(\chi)$ είναι πρωταρχικό πολυώνυμο στο $\mathbb{F}_\rho$ με $\deg(m_s) = n$.

iv. Όλα τα ανάγωγα πολυώνυμα στο $\mathbb{F}_\rho$, ο βαθμός των οποίων διαιρεί το n , έχουν γινόμενο ίσο με $\chi^{\rho^n} - \chi = \chi \prod_{s\in I} m_s(\chi)$

v. Ένα πεπερασμένο σώμα $\mathbb{F}_{\rho^n}$ περιέχει ανάγωγα πολυώνυμα βαθμού n, το πλήθος των οποίων ισούται με: $\frac{1}{n}\sum_{d/n} \mu(n/d)\rho^d$

όπου $\mu(\cdot)$ η συνάρτηση Mobius η οποία δίνεται από τον τύπο

$$\mu(n) = \begin{cases} 0\,, \text{αν ένα τετράγωνο διαιρεί το } n \\ (c-1)^{v(n)}\,, \text{αν ο } n \text{ είναι ελεύθερος τετραγώνου} \end{cases}$$

όπου $v(n) =$ ο αριθμός των διακεκριμένων πρώτων παραγόντων του n.

Παράδειγμα1.4.8 Κατασκευάζουμε τον $\mathbb{F}_{2^4}$ όπως στο παράδειγμα 1.3.5 από το $\mathbb{F}_2$ με τη βοήθεια του ανάγωγου πολυωνύμου $f(x)=x^2+x+1 \in \mathbb{F}_2[\chi]$ που έχει ρίζα το α. Τότε για τις κυκλοτομικές κλάσεις και τα ελάχιστα πολυώνυμα του $\mathbb{F}_{2^4}$ έχουμε:

Στοιχείο	Κυκλοτομική κλάση	Ελάχιστο πολυώνυμο
1	$C_0 = \{0\}$	$m_0(\chi)=\chi+1$
$\alpha,\alpha^2,\alpha^4,\alpha^8$	$C_1=\{1,2,4,8\}$	$m_1(\chi)=\chi^4+\chi+1$
$\alpha^3,\alpha^6,\alpha^{12},\alpha^9$	$C_3=\{3,6,12,9\}$	$m_3(\chi)=\chi^4+\chi^3+\chi^2+\chi+1$
α^5,α^{10}	$C_5=\{5,10\}$	$m_5(\chi)=\chi^2+\chi+1$
$\alpha^7,\alpha^{14},\alpha^{13},\alpha^{11}$	$C_7=\{7,14,13,11\}$	$M_7(\chi)=\chi^4+\chi^3+1$

Επιπλέον $\chi^{2^4} - \chi = \chi \cdot m_0(\chi)m_1(\chi)m_3(\chi)m_5(\chi)m_7(\chi) =$

$$\chi(\chi+1)(\chi^4+\chi+1)(\chi^4+\chi^3+\chi^2+\chi+1)(\chi^2+\chi+1)(\chi^4+\chi^3+1)$$

Οι επικεφαλείς κλάσεων $\mathbb{F}_{2^4}$ είναι $I=\{0,1,3,5,7\}$

Αφού $\alpha^{15} = 1$ τότε $\alpha^{14} = \alpha - 1$

Τότε όμως $m_7(\chi) = \chi^4 m_1(1/\chi)$,όπου $m_1(\chi)$ το ελάχιστο πολυώνυμο του α και $m_7(\chi)$ το ελάχιστο πολυώνυμο του $\alpha^{14}= \alpha^{-1}$ και άρα το $m_7(\chi)$ είναι το αντίστροφο πολυώνυμο του $m_1(\chi)$.

1.5 Υποσώματα

Έχουμε ήδη αναφερθεί στα υποσώματα από την 1.1.2. Στην παρούσα ενότητα εξετάζονται σχέσεις πάνω σε σώματα και η μετάβαση αυτών στα υποσώματά τους. Οι σχέσεις αυτές είναι απαραίτητες στον σχεδιασμό των ακολουθιών οι οποίες παράγονται από καταχωρητές ολίσθησης καθώς επίσης και στο χαρακτηρισμό τους.

Λήμμα 1.5.1 Έστω $\mathrm{n}, \mathrm{m} \in \mathbb{Z}$ με $\mathrm{n}, \mathrm{m} \geq 1$ και $\alpha \in \mathbb{Z}$ με $\alpha \geq 2$, τότε:
$\mu\kappa\delta(\alpha^n - 1, \alpha^m - 1) = \alpha^{\mu\kappa\delta(n,m)} - 1$

Πόρισμα 1.5.2 Έστω οι $\mathrm{N}, \mathrm{M} \in \mathbb{N}$ $\mu\varepsilon$ $N = 2^n - 1, M = 2^m - 1,$ όπου οι $\mathrm{n}, \mathrm{m} \in \mathbb{Z}$ με $\mathrm{n}, \mathrm{m} \geq 1$ τότε: $\mu\kappa\delta(N, M) = 1 \Leftrightarrow \mu\kappa\delta(n, m) = 1$

Θεώρημα 1.5.3 Έστω δύο πεπερασμένα σώματα $F = \mathbb{F}_{\rho^n}$ και $L = \mathbb{F}_{\rho^m}$, με ρ^n και ρ^m πλήθος στοιχείων αντίστοιχα, όπου ρ πρώτος, τότε:

i. Έστω $\chi \in F$, τότε $\chi \in L \Leftrightarrow \chi^{\rho^m} = \chi$.
ii. .Το F περιέχει το L ως υπόσωμα $\Leftrightarrow m/n$.

Πόρισμα 1.5.4 Έστω $F = \mathbb{F}_{\rho^n}$ και $\alpha \in F$ το οποίο έχει ελάχιστο πολυώνυμο $m_\alpha(\chi)$, τότε

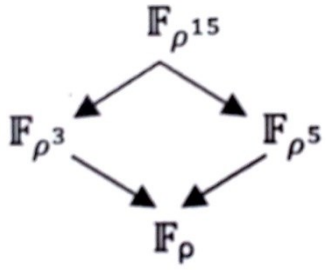
i. $\deg(m)\,/n.$
ii. Αν r η τάξη του α, τότε $\mathrm{r}/\rho^n - 1.$
iii. .$\mathrm{m}_\alpha(\chi)\,/\chi^r - 1$, το οποίο είναι το διώνυμο ελαχίστου βαθμού με αυτήν την ιδιότητα.

Παρατηρήσεις 1.5.5

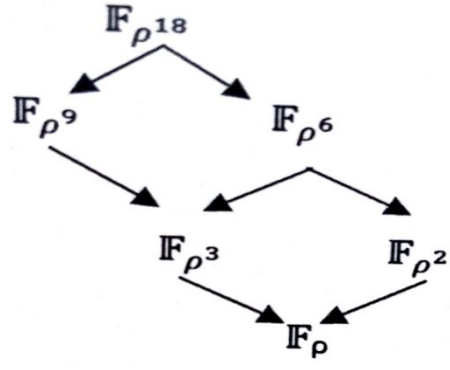
i. Αν n πρώτος, τότε για κάθε στοιχείο του $\mathbb{F}_{\rho^n}$ ο βαθμός του ελάχιστου πολυωνύμου του ισούται με n.
ii. Ως τάξη του $m_\alpha(\chi)$ ορίζουμε το r.

Παράδειγμα1.5.5

i. Τα υποσώματα του $\mathbb{F}_{\rho^{15}}$ καθορίζονται από τους διαιρέτες του 15, ως εξής:



ii. Ενώ για το $\mathbb{F}_{\rho^{18}}$



1.6 Απεικονήσεις σε πεπερασμένα σώματα

Στην παρούσα ενότητα θα ορίσουμε την απεικόνιση του ίχνους και την απεικόνιση της νόρμας από ένα πεπερασμένο σώμα σε ένα πεπερασμένο υπόσωμα του και θα παρουσιάσουμε τις βασικές τους ιδιότητες

1.6.1 Συναρτήσεις ίχνους

Ορισμός 1.6.1 Έστω $K = \mathbb{F}_q, F = \mathbb{F}_{q^n}$, και $\alpha \in F$ το **ίχνος** $Tr_{F/K}(\alpha)$ ενός στοιχείου α στο K ορίζεται από τη σχέση $Tr_{F/K}(\alpha) = \alpha + \alpha^q + \cdots + \alpha^{q^{n-1}}$.

Δηλαδή το ίχνος του α είναι το άθροισμα όλων των συζυγών στοιχείων του α στο $\mathbb{F}_q$.

Αν επιπλέον το K είναι κύριο υπόσωμα του F, το $Tr_{F/K}(\alpha)$ λέγεται απόλυτο ίχνος του α και συμβολίζεται με $Tr_F(\alpha)$(ή $Tr(\alpha)$).

Παράδειγμα 1.6.2 Έστω $K = \mathbb{F}_2$ και $F = \mathbb{F}_{2^4}$,τότε $Tr_{F/K}(\alpha) = \alpha + \alpha^2 + \alpha^4 + \alpha^8$

Ενώ για $K = \mathbb{F}_{2^2}$ και $F = \mathbb{F}_{2^4}$, έχουμε $Tr_{F/K}(\alpha) = \alpha + \alpha^4$

Παρατήρηση 1.6.3

Αφού $((Tr_{F/K}\ (\alpha))^q = \sum_{i=0}^{n-1} \alpha^{q^{i+1}} = Tr_{F/K}\ (\alpha)$, τότε $\forall\ \alpha \in F\ \ Tr_{F/K}\ (\alpha) \in K$

Άρα η συνάρτηση ίχνους $Tr_{F/K}$ είναι μια απεικόνιση από το F στο υπόσωμα K

$Tr_{F/K}\ : F \to K$

Σχόλιο Σύμφωνα με όσα έχουμε δει το q είναι το πλήθος στοιχείων του $\mathbb{F}_q$ και είτε είναι πρώτος αριθμός είτε μπορεί να γραφεί ως δύναμη πρώτου αριθμού.

Θεώρημα 1.6.4 Έστω $K = \mathbb{F}_q$ και $F = \mathbb{F}_{q^n}$ τότε η συνάρτηση ίχνους έχει τις ακόλουθες ιδιότητες:

i. $\forall\ \alpha \in F\ ,\ Tr_{F/K}\ (\alpha) \in K.$

ii. $.\forall\ \alpha, \beta \in F\ ,\ Tr_{F/K}\ (\alpha + \beta) = Tr_{F/K}\ (\alpha) + Tr_{F/K}\ (\beta).$

iii. $.\forall\ \alpha \in F\ και\ c \in K\ ,\ Tr_{F/K}\ (c\alpha) = cTr_{F/K}\ (\alpha).$

iv. Θεωρώντας τα F, K ως διανυσματικούς χώρους στο K , η $Tr_{F/K}$ είναι γραμμικός μετασχηματισμός από το σώμα F επί το υπόσωμα K.

v. $\forall\ \alpha \in K\ \ Tr_{F/K}\ (\alpha) = n\alpha.$

vi. $\forall\ \alpha \in F\ \ Tr_{F/K}\ (\alpha^q) = Tr_{F/K}\ (\alpha).$

vii. $\forall\ \alpha \in K\ \ \left|\{\beta \in F / Tr_{F/K}\ (\beta) = \alpha\}\right| = q^{n-1}.$

viii. Έστω K⊆F⊆E πεπερασμένα σώματα, τότε $\forall\ \alpha \in F$ $Tr_{E/K}\ (\alpha) = Tr_{F/K}\ (Tr_{E/F}\ (\alpha))$

1.6.2 Νόρμες

Ορισμός 1.6.5 Έστω $K = \mathbb{F}_q$, $F = \mathbb{F}_{q^n}$ και $\alpha \in F$, τότε η νόρμα $N_{F/K}\ (\alpha)$ ή $Norm_{F/K}\ (\alpha)$ του στοιχείου α στο K ορίζεται ως: $N_{F/K}\ (\alpha) = \alpha \cdot \alpha^q \cdot \ldots \cdot \alpha^{n-1} = \alpha^{\frac{q^n-1}{q-1}}$

Είναι δηλαδή το γινόμενο όλων των συζυγών του στοιχείου $\alpha \in \mathbb{F}_q$.

Θεώρημα 1.6.6 Έστω $K = \mathbb{F}_q$ και $F = \mathbb{F}_{q^n}$, τότε η συνάρτηση της νόρμας έχει τις ακόλουθες ιδιότητες:

i. $\forall\ \alpha \in F\ ,\ N_{F/K}\ (\alpha) \in K.$

ii. $\forall\ \alpha, \beta \in F\ ,\ N_{F/K}\ (\alpha\beta) = N_{F/K}\ (\alpha) \cdot N_{F/K}\ (\beta).$

iii. Η συνάρτηση νόρμας απεικονίζει το F επί το K και το F^* επί το K^*.

iv. $\forall\ \alpha \in K,\ N_{F/K}(\alpha) = \alpha^n$.

v. $\forall\ \alpha \in F,\ N_{F/K}(\alpha^q) = N_{F/K}(\alpha)$.

vi. Έστω K⊆F⊆E πεπερασμένα σώματα ,τότε $N_{E/K}(\alpha) = N_{F/K}(N_{E/F}(\alpha))$.

1.7 Δυϊκές Βάσεις

Με τη βοήθεια της συνάρτησης ίχνους που ορίσαμε στην 1.6 θα οριστεί στην παρούσα ενότητα η έννοια των δυικών βάσεων του $\mathbb{F}_{q^n}$ στο $\mathbb{F}_q$.

Ορισμός 1.7.1 Έστω $K = \mathbb{F}_q$ και $F = \mathbb{F}_{q^n}$, δύο διατεταγμένες βάσεις του $\mathbb{F}_{q^n}$ στο $\mathbb{F}_q$ $\{\alpha_1, \dots, \alpha_n\}$ και $\{\beta_1, \dots, \beta_n\}$ θα λέγονται δυϊκές αν $\forall\ 1 \le i,j \le n$

$$Tr_{\mathbb{F}_q{}^n/\mathbb{F}_q}(\alpha_i\ \beta_j) = \begin{cases} 1, & \alpha\nu\ i = j \\ 0, & \alpha\nu\ i \neq j \end{cases}$$

Θεώρημα 1.7.2 Έστω $K = \mathbb{F}_q$, $F = \mathbb{F}_{q^n}$, και $\{\alpha_1, \dots, \alpha_n\}$, $\{\beta_1, \dots, \beta_n\}$ δύο δυϊκές βάσεις του Fστο K.

Αν $\chi = \sum_{i=1}^{n} \chi_i\, \alpha_i \in F$, όπου $\chi_i \in K\ \forall\ 1 \le i \le n$, τότε $\chi_i = Tr_{F/K}(\beta_i\, \chi)$

1.8 Συστήματα και ακολουθίες καταγραφής μετατόπισης

Στην παρούσα ενότητα θα παρουσιαστεί η έννοια των συστημάτων καταγραφής μετατόπισης και των ακολουθιών καταγραφής μετατόπισης. Ακολουθιών δηλαδή που παράγονται από ΓΨΑ σε καταχωρητές ολίσθησης με ανάδραση.

Έστω ότι έχουμε n καταγραφείς $R_0, R_1, \dots, R_{n-1}$,όπου $\forall\ i = 0,1,2,\dots,n-1$ οι τιμές των R_ι είναι στο $\mathbb{F}_q$.

Θέτουμε το περιεχόμενο του καταγραφέα R_ι να είναι το $Xi(t), i = 0,1,2,\dots.n-1$.

Και η κατάσταση του συστήματος (των καταγραφέων) τη χρονική στιγμή t να είναι:

$$X(t) = (X_0(t), X_1(t), \dots, X_{n-1}(t)).$$

Ως αρχική κατάσταση θεωρούμε την $X(0) = (X_0(0), X_1(0), \dots, X_{n-1}(0)) \neq (0,0,\dots,0)$.

Τότε ένα σύστημα καταγραφής μετατόπισης με ανάδραση έχει τη μορφή:

$$\begin{cases} X_i(t+1) = X_{i+1}(t), & 0 \le i \le n-2 \\ X_{n-1}(t+1) = h(X_0(t), \dots, X_{n-1}(t)) \end{cases}$$

όπου h: $\mathbb{F}_q^n \to \mathbb{F}_q$ μια συνάρτηση. (Αν η h είναι γραμμική θα έχουμε γραμμικό σύστημα, ενώ αν είναι μη-γραμμική θα έχουμε μη-γραμμικό σύστημα) Στις περισσότερες περιπτώσεις ο σταθερός όρος της h ισούται με 0.

Επεξήγηση Ο τρόπος λειτουργίας του συστήματος είναι ο ακόλουθος: κάθε χρονική στιγμή t ο καταγραφέας R_{n-1} δέχεται ως περιεχόμενο την τιμή της συνάρτησης h, το περιεχόμενο του καταγραφέα R_i στέλνετε στον R_{i-1} και το περιεχόμενο του καταγραφέα R_0 στέλνετε στην έξοδο.

Ορισμός 1.8.1 Μια ακολουθία $(\beta_i)_{i=0}^{\infty}$ με στοιχεία σε ένα σώμα $\mathbb{F}_q$ ονομάζεται τελικά περιοδική αν υπάρχουν ακέραιοι $T > 0$ και $t_0 \geq 0$ τέτοιοι ώστε

$$\beta_{i+T} = \beta_i, \forall i \geq t_0$$

Ο μικρότερος T με την παραπάνω ιδιότητα ονομάζεται πρωταρχική περίοδος της ακολουθίας $(\beta_i)_{i=0}^{\infty}$ και ο ακέραιος t_0 είναι η προ-περίοδος της $(\beta_i)_{i=0}^{\infty}$.
Αν $t_0 = 0$ η ακολουθία λέγεται περιοδική.

► Θεωρούμε την ακολουθία εξόδου του συστήματος (α_i) , με $\alpha_i = \mathrm{X}_0(\mathrm{i})$, $i \in \mathbb{N}$ και έχουμε:

$$\alpha_{i+n} = \mathrm{X}_0(\mathrm{i}+\mathrm{n}) = \mathrm{X}_1(\mathrm{i}+\mathrm{n}-1) = \cdots = \mathrm{X}_{\mathrm{n}-1}(\mathrm{i}+1) = \mathrm{X}_{\mathrm{n}}(i) = h\big(X_0(i), \dots, X_{n-1}(i)\big) =$$
$$= h(\alpha_i, \alpha_{i+1}, \dots, \alpha_{i+n-1}), i \geq 0$$

Μια τέτοια ακολουθία θα την ονομάζουμε ακολουθία καταγραφής μετατόπισης και θα είναι (τελικά) περιοδική.

Για κάθε χρονική στιγμή $i \geq 0$ η κατάσταση του συστήματος δίνεται από το διάνυσμα $(\alpha_{i+n-1}, \alpha_{i+n-2}, \dots, \alpha_i)$.
Για μήκος n μπορούμε να έχουμε q^n διαφορετικές καταστάσεις. Άρα η μέγιστη περίοδος που μπορεί να έχει η θεωρητικά η (α_i) είναι q^n, τότε η περίοδος της περιέχει όλες τις δυνατές $n-$άδες στο $\mathbb{F}_q$ και ονομάζεται ακολουθία De Brujin. Αν όμως περιοριστούμε σε συναρτήσεις ανάδρασης h με μηδενικό σταθερό όρο (που είναι αυτές που μας ενδιαφέρουν κυρίως), τότε η μέγιστη δυνατή περίοδος (εξόδου) είναι $q^n - 1$(αυτό συμβαίνει επειδή αν το σύστημα περάσει από την μηδενική κατάσταση θα παραμείνει για πάντα σε αυτήν).

Σχόλιο Στις κρυπτογραφικές εφαρμογές, συνήθως, δε δουλεύουμε γενικά σε ένα πεπερασμένο σώμα $\mathbb{F}_q$, αλλά στο $\mathbb{F}_2 = \{0,1\}$ και οι συναρτήσεις h που χρησιμοποιούμε είναι συνήθως σταθερού μηδενικού όρου.

2 Κριτήρια αξιολόγησης ψευδοτυχαίων ακολουθιών

Εισαγωγή

Στο κεφάλαιο αυτό θα παρουσιαστεί η σουίτα Στατιστικών ελέγχων (tests) για γεννήτριες τυχαίων και ψευδοτυχαίων (δυαδικών) ακολουθιών (πεπερασμένου μήκους) για Κρυπτογραφικές εφαρμογές (A Statistical Test Suite For Random and Pseudorandom Number Generators for Cryptographic Applications), που αναπτύχθηκε από το Εθνικό Ινστιτούτο Προτύπων και Τεχνολογίας των Η.Π.Α (National Institute of Standards and Technology -NIST-) το 2001 και έχει υποστεί πληθώρα αναθεωρήσεων έκτοτε, με τελευταία τον Απρίλιο του 2010, στην οποία και θα αναφερθούμε Είναι ένα λογισμικό πακέτο αποτελούμενο από μια σειρά 15 ελέγχων (tests). Οι οποίοι έχουν ως σκοπό να εξετάσουν ως προ την τυχαιότητα τους δυαδικές ακολουθίες που έχουν παραχθεί από Γ.Τ.Α ή Γ.Ψ.Α και κατ' επέκταση των γεννητριών που τις παράγουν, προοριζόμενες (κυρίως) για κρυπτογραφικές εφαρμογές. Έτσι οι υπό εξέταση ακολουθίες ελέγχονται ως προς μια πληθώρα τύπων μη-τυχαιότητας, που θα μπορούσαν να έχουν. Κάθε έλεγχος εξετάζει την ακολουθία που μας ενδιαφέρει και παράγει μια στατιστική τιμή, που την αποκαλεί $P-value$, η οποία αντιπροσωπεύει την πιθανότητα μια τέλεια Γ.Ψ.Α (ή Γ.Τ.Α) να παράγει μια ακολουθία ¨λιγότερο τυχαία¨ από την εξεταζόμενη. Αν, δηλαδή, έχουμε ως αποτέλεσμα ενός τεστ $P-value = 0.01$, σημαίνει ότι το 1% των ακολουθιών που παράγει μια τέλεια ΓΨΑ, θα δείχνουν ¨λιγότερο τυχαίες¨ από αυτήν που εξετάζεται. Όπως γίνεται εύκολα αντιληπτό αν $P-value = 0$ η ακολουθία μπορεί να θεωρηθεί ως ¨πλήρως μη-τυχαία¨, ενώ αν $P-value = 1$ η ακολουθία μπορεί να θεωρηθεί ότι έχει ¨την τέλεια τυχαιότητα¨. Όλοι οι έλεγχοι απαιτούν να έχει οριστεί εκ των προτέρων μια τιμή ως όριο εμπιστοσύνης (α). Έτσι απαιτείται $P-value \geq \alpha$, για να θεωρήσουν την υπό εξέταση ακολουθία (θα καλείται καταχρηστικά ε) ως τυχαία και να έχουμε επιτυχία στο τεστ, αλλιώς έχουμε αποτυχία. Το όριο εμπιστοσύνης α αντιστοιχεί στην πιθανότητα να απορριφθεί μια ακολουθία ως μη τυχαία ενώ στην πραγματικότητα είναι τυχαία. Συνήθως $\alpha \in [0.001, 0.01]$. Η σουίτα έχει προεπιλεγμένη επιλογή $\alpha = 0.01$, αυτό σημαίνει ότι αναμένουμε να απορρίπτεται 1 στις 100 ακολουθίες που παράγονται από μια καλή (τυχαία) γεννήτρια και τότε η σχέση $P-value \geq 0.01$ σημαίνει ότι κατά 99% η υπό εξέταση ακολουθία είναι τυχαία. Δεν υπάρχει συγκεκριμένη σειρά εκτέλεσης των ελέγχων, όμως συνιστάται να εκτελείται πρώτο το Frequency (Monobit) Test (2.1), αφού εξετάζει θεμελιώδη στοιχεία για την ύπαρξη τυχαιότητας μιας ακολουθίας και αν αποτύχει η πιθανότητα αποτυχίας είναι πολύ υψηλή και για τα υπόλοιπα τεστ. Όπως θα παρουσιαστεί, κάθε έλεγχος χρησιμοποιεί κάποια βασική κατανομή ως κατανομή αναφοράς. Για παράδειγμα χρησιμοποιεί την κατανομή X^2 για να συγκρίνει τις συχνότητες της με αυτές που παρατηρήθηκαν στην ε, έτσι ώστε να μπορέσει να αποφανθεί. Επειδή αναφερόμαστε σε δυαδικές ακολουθίες που αποσκοπούν να

χρησιμοποιηθούν σε κρυπτογραφικές εφαρμογές, κάθε στοιχείο της ε θα ανήκει στο πεπερασμένο σώμα $\mathbb{Z}_2$ και για το πρόγραμμα θα αναγνωρίζεται ως bit. Υπάρχουν διαφορετικές απαιτήσεις από το κάθε τεστ ως προς τα μεγέθη που εισάγονται και χρησιμοποιεί, αφενός για να υπάρχει νόημα στους ελέγχους που εκτελούνται και αφετέρου γιατί αναφερόμαστε σε προγράμματα των οποίων ο κώδικας έχει γραφεί με τρόπο τέτοιο ώστε να μπορούν να ανταποκριθούν όσο καλύτερα γίνεται σε πρακτικές προκλήσεις, επιτρέποντας έτσι θεωρητικές αδυναμίες (σύνηθες φαινόμενο στις υπολογιστικές εφαρμογές γενικότερα). Μια από τις συχνότερες απαιτήσεις που παρουσιάζονται αφορά στο μέγεθος της υπό εξέταση ακολουθίας (πλήθος στοιχείων-bits), απαίτηση την οποία θα παραβλέψουμε στην πλειονότητα των παραδειγμάτων που θα παρουσιαστούν στο παρόν κεφάλαιο, τα οποία έχουν ως σκοπό να παρουσιάσουν/επεξηγήσουν τον τρόπο λειτουργίας του κάθε ελέγχου στον αναγνώστη, πράγμα που δεν θα μπορούσε να γίνει πρακτικά σε γραπτό κείμενο για ακολουθίες που πληρούν την άνω απαίτηση. Θα παρουσιαστούν, όμως αποτελέσματα/παραδείγματα των ελέγχων για ακολουθίες που πληρούν τις απαιτήσεις στο Κεφάλαιο 4.

2.1 Frequency (Monobit) Test

Είναι ο πρώτος έλεγχος που πρέπει να εκτελέσουμε και σε περίπτωση αποτυχίας δεν έχει νόημα να συνεχίσουμε με τους υπόλοιπους. Σκοπός του τεστ είναι να εξετάσει την αναλογία των άσσων και των μηδενικών της παραγόμενης ακολουθίας. Θέλουμε να δούμε, δηλαδή, κατά πόσο το πλήθος των άσσων και των μηδενικών που εμφανίζονται σε όλο το μήκος της ακολουθίας είναι κατά προσέγγιση αυτός που θα αναμέναμε από μια πραγματικά τυχαία ακολουθία. Πιο συγκεκριμένα επικεντρωνόμαστε στο ποσοστό εμφάνισης των άσσων και εξετάζουμε πόσο κοντά είναι αυτό στο $\frac{1}{2}$, γεγονός που θα σημαίνει ότι άσσοι και μηδενικά θα έχουν περίπου ίδιο αριθμό εμφανίσεων. Αν το τεστ επιτύχει θα πρέπει να προχωρήσουμε στα επόμενα τεστ, για να μπορέσουμε να αποφανθούμε για την τυχαιότητα της ακολουθίας μας. Το τεστ χρησιμοποιεί ως κατανομή αναφοράς την ημι-κανονική κατανομή (η μία της πλευρά είναι διάφορη της κανονικής κατανομής).

Δεδομένα:

1. n=το μήκος της υπό εξέταση ακολουθίας.
2. $\varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n) =$ η υπό εξέταση ακολουθία.
   (Συνιστώμενα μεγέθη: $n \geq 100$).

Διαδικασία:

1. $S_n = \chi_1 + \chi_2 + \cdots + \chi_n$,όπου χ_i=2ε_i-1, $i = 1, \ldots, n$
   (Μετατρέπονται τα στοιχεία της ακολουθίας ε που είναι ίσα με 0 σε -1, ενώ αυτά που είναι ίσα με 1 παραμένουν ως έχουν και παράγεται έτσι μια νέα ακολουθία $(\chi_n)_{n=1}^{\infty} = X$, της οποίας υπολογίζεται το άθροισμα S_nόλων των ψηφίων).
2. $S_{obs} = \frac{|S_n|}{\sqrt{n}}$
3. $P - value = erfc(\frac{S_{obs}}{\sqrt{2}})$, όπου $erfc(\cdot)$ είναι μια συμπληρωματική συνάρτηση σφάλματος που ορίζεται ως: $erfc(z) = \frac{2}{\sqrt{\pi}} \int_z^{\infty} e^{-u^2} du$

Συμπέρασμα: Η ε δεν είναι τυχαία και αποτυγχάνει στο τεστ αν $P - value < 0.01$. Αλλιώς (αν $P - value \geq 0.01$) η ε είναι τυχαία.

Σημείωση: Για να έχουμε $P - value < 0.01$ και να αποτύχει το τεστ θα έχουμε ότι η τιμή του $|S_n|$ ή του $|S_{obs}|$ θα είναι πολύ μεγάλη. $S_n \gg 0$ σημαίνει πολλά 1, ενώ $S_n \ll 0$ σημαίνει πολλά 0.

Παράδειγμα 2.1.1

Για $\varepsilon = 1011010101, n = 10$

(βήμα 1^o)

$S_n = 1 + (-1) + 1 + 1 + (-1) + 1 + (-1) + 1 + (-1) + 1 = 2$

(βήμα 2^o)

$S_{obs} = \frac{|2|}{\sqrt{10}} = 0.6324553$

(βήμα 3^o)

$P - value = erfc\left(\frac{0.63245532}{\sqrt{2}}\right) = 0.527089$

2.2 Frequency Test within a Block

Σκοπός του τεστ είναι να εξετάσει αν το πλήθος των άσσων (άρα και των μηδενικών-όπως είπαμε και πριν) σε ένα μπλοκ μήκους M bits της υπό εξέταση ακολουθίας είναι αυτός που θα αναμέναμε από μια τυχαία ακολουθία (κατά προσέγγιση $M/2$). Για block μήκους 1 (M=1) το τεστ εκφυλίζεται στο Frequency monobit τεστ. Το τεστ χρησιμοποιεί ως κατανομή αναφοράς την κατανομή X^2 .

Δεδομένα:

1. $M =$ το μήκος κάθε μπλοκ.
2. $n =$ το μήκος της υπό εξέταση ακολουθίας.
3. $\varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n) =$ η υπό εξέταση ακολουθία.
   (Συνιστώμενα μεγέθη: $n \geq 100$, $M \geq 10, M > 0.01 \cdot n$ -για να έχουμε $N < 100$, αφού $n \geq MN$-)

Διαδικασία:

1. $N = floor(\frac{n}{M}) =$ το πλήθος των blocks που θα δημιουργηθούν (όσα bits περισσεύουν απορρίπτονται), όπου $floor(x) = max\{m \in \mathbb{Z} / m \leq x\}$
2. $\pi_i = \frac{\sum_{j=1}^{M} \varepsilon_{(i-1)M+j}}{M}, για\ i = 1,2, \dots, N$
   (υπολογίζεται η αναλογία των 1 σε κάθε block μήκους M)
3. $\chi^2(obs) = 4M \sum_{i=1}^{N} (\pi_\iota - \frac{1}{2})^2$,
   (η $\chi^2(obs)$ αποτελεί ένα μέτρο για το αν η αναλογία των 1 σε ένα M-block πλησιάζει στην επιθυμητή αναλογία του $\frac{1}{2}$)
4. $P - value = igamc(\frac{N}{2}, \frac{X^2(obs)}{2})$, όπου $igamc(\cdot)$ είναι η ημιτελείς συνάρτηση Γάμμα που ορίζεται ως εξής:
   $igamc(\alpha, \chi) = \frac{\Gamma(\alpha, \chi)}{\Gamma(\alpha)} = \frac{1}{\Gamma(\alpha)} \int_{\chi}^{\infty} e^{-t} t^{\alpha-1} dt$,
   (με $igamc(\alpha, 0) = 1\ και\ igamc(\alpha, \infty) = 0$)
   και $\Gamma(\alpha)$ η συνάρτηση Γάμμα που ορίζεται ως: $\Gamma(\alpha) = \int_0^{\infty} e^{-t} t^{\alpha-1} dt$

Συμπέρασμα: Η ε δεν είναι τυχαία και αποτυγχάνει στο τεστ αν $P - value < 0.01$. Αλλιώς (αν $P - value \geq 0.01$) η ε είναι τυχαία.

Σημείωση: Για να έχουμε ότι $P - value < 0.01$ και να αποτύχει το τεστ θα έχει προκληθεί από μεγάλη απόκλιση από την ίση αναλογία 1 και 0, τουλάχιστον σε ένα block.

Παράδειγμα 2.2.1

Για $\varepsilon = 0110011010, n = 10, M = 3$

(βήμα 1^o)

$N = floor(10/3) = 3$, άρα τα 3 blocks που θα δημιουργηθούν είναι: 011, 001 και 101.

Το τελευταίο 0 της ε απορρίπτεται.

(βήμα 2^o)

$\pi_1 = \frac{2}{3}, \pi_2 = \frac{1}{3}$ και $\pi_3 = \frac{2}{3}$

(βήμα 3^o)

$\chi^2(obs) = 4 \cdot 3 \cdot \left[\left(\frac{2}{3} - \frac{1}{2}\right)^2 + \left(\frac{1}{3} - \frac{1}{2}\right)^2 + \left(\frac{2}{3} - \frac{1}{2}\right)^2\right] = 1$

(βήμα 4^o)

$P - value = igamc(\frac{3}{2}, \frac{1}{2}) = 0.801252$

2.3 Runs Test

Ο έλεγχος αυτός εξετάζει τον συνολικό αριθμό σειρών-ροών των άσσων και μηδενικών. Με τον όρο σειρά (ή αλλιώς ροή) αναφερόμαστε σε κάθε υπακολουθία (πεπερασμένου μήκους) που αποτελείται από διαδοχικά όμοια στοιχεία-bits (0 ή 1). Μια σειρά μήκους κ αποτελείται από κ το πλήθος διαδοχικά όμοια στοιχεία και οριοθετείται πριν και μετά από ένα στοιχείο με αντίθετη τιμή. Το τεστ αποσκοπεί στο να καθαρίσει αν το πλήθος των (διαφόρων μηκών) ροών 0 και 1, είναι το αναμενόμενο για μια τυχαία ακολουθία. Εξετάζεται, δηλαδή, αν η ταλάντωση μεταξύ 0 και 1 είναι πολύ αργή ή πολύ γρήγορη. Προϋπόθεση του τεστ είναι να έχει εκτελεστεί πρώτα το frequency test. Το τεστ χρησιμοποιεί ως κατανομή αναφοράς την κατανομή X^2.

Δεδομένα

1. :n=το μήκος της υπό εξέταση ακολουθίας.
2. $\varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n) =$ η υπό εξέταση ακολουθία.
   (Συνιστώμενα μεγέθη: $n \geq 100$).

Διαδικασία:

1. $\pi = \sum_j \frac{\varepsilon_j}{n}$
   (υπολογίζεται η αναλογία των 1 στην ακολουθία και αποτελεί pre-test μέγεθος που θα μας βοηθήσει να δούμε αν η ακολουθία έχει περάσει το τεστ)
2. Αν $|\pi - 1/2| \geq \tau$, όπου $\tau = \frac{2}{\sqrt{n}}$, τότε $P - value = 0.0000$
   (το pre-test αποτυγχάνει αφού η ακολουθία θα πρέπει να έχει ήδη αποτύχει στο 2.1)
   Αλλιώς βήμα 3 (προχωράμε στο runs test).
3. $v_n(obs) = \sum_{\kappa=1}^{n-1} r(\kappa) + 1$, όπου: αν $\varepsilon_\kappa = \varepsilon_{\kappa+1}$ $r(\kappa) = 0$, αλλιώς $r(\kappa) = 1$
   (υπολογίζεται το συνολικό πλήθος ροών -και των 1 και των 0- της ακολουθίας)
4. $P - value = erfc(\frac{|v_n(obs) - 2n\pi(1-\pi)|}{2\sqrt{2n}\pi(1-\pi)})$

Συμπέρασμα: Η ε δεν είναι τυχαία και αποτυγχάνει στο τεστ αν $P-value < 0.01$. Αλλιώς (αν $P-value \geq 0.01$) η ε είναι τυχαία.

Σημείωση: Μια μεγάλη τιμή του $v_n(obs)$ σημαίνει πολύ γρήγορη ταλάντωση στην ακολουθία, ενώ μια μικρή τιμή πολύ αργή ταλάντωση (πράγμα που σημαίνει μικρότερο πλήθος ροών από ότι σε μια τυχαία ακολουθία).

Παράδειγμα 2.3.1

Για ε=1001101011, n=10

(βήμα 1^o)

$\pi=\frac{6}{10}=\frac{3}{5}$

(βήμα 2^o)

$\tau=\frac{2}{\sqrt{10}}=0.63246$, τότε $\left|\pi-\frac{1}{2}\right|=\left|\frac{3}{5}-\frac{1}{2}\right|=0.1<\tau$ άρα μπορούμε να προχωρήσουμε στο Runs Test

(βήμα 3^o)

$v_{10}(obs)=(1+0+1+0+1+1+1+1+0)+1=7$

(βήμα 4^o)

$$P-value=erfc\left(\frac{\left|7-2\cdot 10\cdot\frac{3}{5}\left(1-\frac{3}{5}\right)\right|}{2\sqrt{20}\cdot\frac{3}{5}\left(1-\frac{3}{5}\right)}\right)=0.147232$$

2.4 Test for the Longest Run of Ones in a Block

Όπως και στο 2.2, το τεστ χωρίζει την ακολουθία σε N blocks μεγέθους M το καθένα για να εξετάσει τη μεγαλύτερη ροή 1 μέσα σε αυτά. Σκοπός του είναι να εξετάσει αν το μήκος της μεγαλύτερης σειράς 1, της υπό εξέταση ακολουθίας, είναι ο αναμενόμενος από μια τυχαία ακολουθία. Αν προκύψει κάποια ανωμαλία όσων αφορά στο μήκος της αναμενόμενης μεγαλύτερης ροής 1, αυτό σημαίνει ότι υπάρχει ανωμαλία και στη μεγαλύτερη ροή 0. Επομένως δεν χρειαζόμαστε ξεχωριστό έλεγχο για τα 0. Το τεστ χρησιμοποιεί ως κατανομή αναφοράς την κατανομή X^2.

Δεδομένα:

1. n=το μήκος της υπό εξέταση ακολουθίας.
2. $\varepsilon=(\varepsilon_1,\varepsilon_2,\ldots,\varepsilon_n)=$ η υπό εξέταση ακολουθία.
3. $M=$ το μήκος κάθε block. Ο αλγόριθμος του τεστ υποστηρίζει τρεις προκαθορισμένες τιμές για το M σε αντιστοιχία με την ελάχιστη τιμή του n, σύμφωνα με τον ακόλουθο πίνακα:

Minimum n	M
128	8
6272	128
750000	10^4

4. N=το πλήθος των blocks (εξαρτάται από το προκαθορισμένο M).

Διαδικασία:

1. Η ακολουθία χωρίζεται σε blocks μεγέθους M.
2. Καταγράφονται σε πίνακές οι συχνότητες v_i των μεγαλύτερων ροών των 1 σε κάθε block, σε κατηγορίες, όπου κάθε κελί περιέχει το πλήθος των μέγιστων ροών των 1 που έχουν το ζητούμενο μήκος σύμφωνα με τον πίνακα:

v_i	M=8	M=128	M=10^4
v_0	≤ 1	≤ 4	≤ 10
v_1	2	5	11
v_2	3	6	12
v_3	≥ 4	7	13
v_4		8	14
v_5		≤ 9	15
v_6			≤ 16

3. $\chi^2(obs) = \sum_{i=0}^{K} \frac{(v_i - N\pi_i)^2}{N\pi_i}$,

   ($\eta\ \chi^2(obs)$ αποτελεί ένα μέτρο για το πόσο καλά η εξεταζόμενη μέγιστη ροή στα blocks μεγέθους M ταιριάζει με αυτήν που θα αναμέναμε από μία τυχαία ακολουθία).

   οι τιμές των K και N καθορίζονται από την τιμή του M σύμφωνα με τον πίνακα:

M	K	N
8	3	16
128	5	49
10^4	6	75

οι τιμές των π_i για τις τιμές των M- άρα και των K- που έχουν ενσωματωθεί στο πρόγραμμα είναι:

Για $K = 3, M = 8$

$\{\nu \leq 1\}$	$\pi_0 = 0.2148$
$\{\nu = 2\}$	$\pi_1 = 0.3672$
$\{\nu = 3\}$	$\pi_2 = 0.2305$
$\{\nu \geq 4\}$	$\pi_3 = 0.1875$

Για $K = 5, M = 128$

$\{\nu \leq 4\}$	$\pi_0 = 0.1174$
$\{\nu = 5\}$	$\pi_1 = 0.2430$
$\{\nu = 6\}$	$\pi_2 = 0.2493$
$\{\nu = 7\}$	$\pi_3 = 0.1752$
$\{\nu = 8\}$	$\pi 4 = 0.1027$
$\{\nu \geq 9\}$	$\pi 5 = 0.1124$

Για $K = 5, M = 512$

$\{\nu \leq 6\}$	$\pi_0 = 0.1170$
$\{\nu = 7\}$	$\pi_1 = 0.2460$
$\{\nu = 8\}$	$\pi_2 = 0.2523$
$\{\nu = 9\}$	$\pi_3 = 0.1755$
$\{\nu = 10\}$	$\pi 4 = 0.1027$
$\{\nu \geq 11\}$	$\pi 5 = 0.1124$

Για $K = 5, M = 1000$

$\{\nu \leq 7\}$	$\pi_0 = 0.1307$
$\{\nu = 8\}$	$\pi_1 = 0.2437$
$\{\nu = 9\}$	$\pi_2 = 0.2452$
$\{\nu = 10\}$	$\pi_3 = 0.1714$
$\{\nu = 11\}$	$\pi_4 = 0.1002$
$\{\nu \geq 12\}$	$\pi_5 = 0.1088$

Για $K = 6, M = 10^4$

$\{\nu \leq 10\}$	$\pi_0 = 0.0882$
$\{\nu = 11\}$	$\pi_1 = 0.2092$
$\{\nu = 12\}$	$\pi_2 = 0.2483$
$\{\nu = 13\}$	$\pi_3 = 0.1933$
$\{\nu = 14\}$	$\pi_4 = 0.1208$
$\{\nu = 15\}$	$\pi_5 = 0.0675$
$\{\nu \geq 16\}$	$\pi_6 = 0.0727$

4. $P - value = igamc(\frac{K}{2}, \frac{\chi^2(obs)}{2})$

Συμπέρασμα: Η ε δεν είναι τυχαία και αποτυγχάνει στο τεστ αν $P - value < 0.01$. Αλλιώς (αν $P - value \geq 0.01$) η ε είναι τυχαία.

Σημείωση: Μεγάλες τιμές της $\chi^2(obs)$ δείχνουν ότι η υποεξέταση ακολουθία έχει πληθώρα 1.

Παράδειγμα 2.4.1

Για
ε=11001100000101010110110001001100111000000000001001001101010100010001001111010110100000001101011111001100111001101101100010110010, n=128, K=3 και M=8
(βήμα 1^o)
υπό-block 11001100, μέγιστη ροή 2
υπό-block 00010101, μέγιστη ροή 1
υπό-block 01101100, μέγιστη ροή 2
υπό-block 01001100, μέγιστη ροή 2
υπό-block 11100000, μέγιστη ροή 3
υπό-block 00000010, μέγιστη ροή 1
υπό-block 01001101, μέγιστη ροή 2
υπό-block 01010001, μέγιστη ροή 1
υπό-block 00010011, μέγιστη ροή 2
υπό-block 11010110, μέγιστη ροή 2
υπό-block 10000000, μέγιστη ροή 1
υπό-block 11010111, μέγιστη ροή 3
υπό-block 11001100, μέγιστη ροή 2
υπό-block 11100110, μέγιστη ροή 3
υπό-block 11011000, μέγιστη ροή 2
υπό-block 10110010, μέγιστη ροή 2
(βήμα 2^o)
$v_0 = 4, v_1 = 9, v_2 = 3, v_3 = 0$
(βήμα 3^o)
$\chi^2(obs) = 4.882457$
(βήμα 4^o)
$P - value = 0.180609 \geq 0.01$, άρα η ε μπορεί να θεωρηθεί τυχαία ακολουθία.

2.5 Binary Matrix Rank test

Αυτό το τεστ επικεντρώνεται στην ταξινόμηση ξένων υπο-πινάκων ολόκληρης της ακολουθίας. Έχοντας ως σκοπό να ελέγξει αν υπάρχει γραμμική εξάρτηση ανάμεσα σε υπακολουθίες προκαθορισμένου μήκους της αρχικής ακολουθίας. Το τεστ χρησιμοποιεί ως κατανομή αναφοράς την κατανομή X^2.

Δεδομένα:

1. n=το μήκος της υπό εξέταση ακολουθίας.
2. $\varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n) =$ η υπό εξέταση ακολουθία.
3. M= το πλήθος των γραμμών κάθε πίνακα. Το πρόγραμμα θέτει $M = 32$, για άλλες τιμές του M πρέπει να ακολουθήσουν άλλοι υπολογισμοί.
4. Q= το πλήθος των στηλών κάθε πίνακα. Το πρόγραμμα θέτει $Q = 32$, για άλλες τιμές του M πρέπει να ακολουθήσουν άλλοι υπολογισμοί. (Συνιστώμενα μεγέθη: πρέπει $n \geq 38MQ$, για να δημιουργηθούν τουλάχιστον 38 πίνακες. Για $M = Q = 32$, πρέπει $n \geq 38912$)

Διαδικασία:

1. Διαιρείται ακολουθιακά η ε σε ξένα blocks μεγέθους $M \cdot Q$, θα έχουμε τότε $N = floor(\frac{n}{MQ})$ τέτοια blocks. Αν περισσεύουν κάποια στοιχεία, αυτά απορρίπτονται. Από τα $M \cdot Q$ μεγέθους τμήματα σχηματίζουμε πίνακες MxQ, έτσι ώστε κάθε γραμμή του πίνακα να αποτελείται από διαδοχικά Q-bit blocks της αρχικής ακολουθίας ε.
2. Προσδιορίζεται η τάξη (rank) του κάθε πίνακα[2] και συμβολίζεται ως R_l, όπου $l = 1,2, \dots, n$
3. Έστω $F_M =$ το πλήθος των πινάκων όπου $R_l = M$ (full rank),
   $F_{M-1} =$ το πλήθος των πινάκων όπου $R_l = M - 1$ (full rank-1),
   $N - F_M - F_{M-1}$= το πλήθος των υπόλοιπων πινάκων.
4. $\chi^2(obs) = \frac{(F_M - 0{,}2888\mathrm{N})^2}{0.2888\mathrm{N}} + \frac{(F_{M-1} - 0.5776\mathrm{N})^2}{0.5776\mathrm{N}} + \frac{(N - F_M - F_{M-1} - 0.1336\mathrm{N})^2}{0.1336\mathrm{N}}$
5. $P - value = e^{-\chi^2(obs)/2}$.

[2] Το πρόγραμμα περιέχει ενσωματωμένο αλγόριθμο εμπρός και πίσω αντικατάστασης, έτσι ώστε να μετατρέψει τον κάθε (δυαδικό) πίνακα -μέσω γραμμοπράξεων που χρησιμοποιούν ως άθροισμα την λογική πύλη XOR- σε τριγωνικό και τότε η τάξη του είναι το πλήθος των μη μηδενικών γραμμών του.

Συμπέρασμα: Η ε δεν είναι τυχαία και αποτυγχάνει στο τεστ αν $P-value < 0.01$. Αλλιώς (αν $P-value \geq 0.01$) η ε είναι τυχαία.

Σημείωση: Μεγάλες τιμές της $\chi^2(obs)$ (και άρα μικρές τιμές της P-value) δείχνουν απόκλιση της κατανομής των τάξεων από αυτήν που θα αναμέναμε για μια τυχαία ακολουθία.

Παράδειγμα 2.5.1

Για $\varepsilon = 01011001001010101101,\ n = 20,\ M = Q = 3$

(βήμα 1^o)

$N = 2$ και οι 2 πίνακες είναι $\begin{pmatrix} 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \end{pmatrix}$ και $\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}$

(βήμα 2^o)

$R_1 = 2$ και $R_2 = 3$

(βήμα 3)

$F_M = F_3 = 1$ και $F_{M-1} = F_2 = 1$

(βήμα 4^o)

$\chi^2(obs) = 0.596953.$

(βήμα 5^o)

$P-value = e^{-0.596953/1} = 0.741948$

2.6 Discrete fourier transform (spectral) test

Αυτός ο έλεγχος επικεντρώνεται στα ύψη των κορυφών του Διακριτού Μετασχηματισμού Fourier (DFT). Έχοντας ως σκοπό να βρει απόκλιση από την τυχαιότητα, για την υπό εξέταση ακολουθία, η οποία προκαλείται από επαναλαμβανόμενα μοτίβα που είναι το ένα κοντά στο άλλο, δηλαδή από περιοδικά χαρακτηριστικά. Ελέγχοντας, ουσιαστικά, αν ο αριθμός των κορυφών που υπερβαίνουν το φράγμα του 95%, είναι σημαντικά διαφορετικός του 5%. Το τεστ χρησιμοποιεί ως κατανομή αναφοράς την κανονική κατανομή.

Δεδομένα:

1. n=το μήκος της υπό εξέταση ακολουθίας.
2. $\varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n) =$ η υπό εξέταση ακολουθία.
   (Συνιστώμενα μεγέθη: $n \geq 1000$)

Διαδικασία:

1. χ_i=2ε_i-1, $i = 1, \dots, n$
   (Μετατρέπονται τα στοιχεία της ακολουθίας ε που είναι ίσα με 0 σε -1, ενώ αυτά που είναι ίσα με 1 παραμένουν ως έχουν και παράγεται έτσι μια νέα ακολουθία $(\chi_n)_{n=1}^{\infty}$=X.)
2. $S = DFT(X)$, όπου $DFT(X) = DFT(\chi_1, .., \chi_n) = (f_0, \dots, f_n) = (f_j)_{j=0}^{n-1}$ με $f_j = \sum_{\kappa=1}^{n} \chi_\kappa e^{2\pi i(\kappa-1)j/n}$ και $e^{2\pi i\kappa j/n} = \cos\left(2\pi i\kappa j/n\right) + isin(2\pi i\kappa j/n)$, (εφαρμόζεται ο Διακριτός Μετασχηματισμός Fourier στην X και παράγεται έτσι μια νέα μιγαδική ακολουθία $S = (s_j)_{j=0}^{n-1}$, η οποία αναπαριστά περιοδικά στοιχεία της υπό-εξέταση ακολουθίας σε διαφορετικές συχνότητες).
3. $M = modulus(S') \equiv |S'|$, όπου $S' = (s_0, \dots, s_{\frac{n}{2}-1})$ η υπακολουθία που αποτελείται από τα $n/2$ πρώτα στοιχεία της S και η συνάρτηση $modulus$ παράγει μια ακολουθία από ύψη κορυφών.
4. $T = \sqrt{(log\frac{1}{0.05})n}$ (Αποτελεί το φράγμα, κατά την υπόθεση της τυχαιότητας, από το οποίο θα πρέπει να είναι μικρότερο το 95% των τιμών που παράγονται από το τεστ).
5. $N_0 = 0.95 \cdot \frac{n}{2}$ (Το πλήθος των κορυφών (95%) όπου θεωρητικά -κατά την υπόθεση της τυχαιότητας- θα αναμέναμε να είναι μικρότερες του T).
6. Υπολογισμός N_1, όπου N_1 το πραγματικό (παρατηρούμενο) πλήθος των κορυφών που είναι μικρότερες του T.
7. $d = \frac{(N_1 - N_0)}{\sqrt{0.95 \cdot 0.05 \cdot n/4}}$
   (Η κανονικοποιημένη διαφορά του αναμενόμενου από το παρατηρούμενο πλήθος κορυφών που είναι κάτω από το όριο).
8. $P - value = erfc(\frac{|d|}{\sqrt{2}})$

Συμπέρασμα: Η ε δεν είναι τυχαία και αποτυγχάνει στο τεστ $αν\ P - value < 0.01$. Αλλιώς (αν $P - value \geq 0.01$) η ε είναι τυχαία.

Σημείωση: Πολύ μικρή τιμή της μεταβλητής d σημαίνει ότι έχουμε πολύ λίγες κορυφές μικρότερες του T (λιγότερες από το 95% των κορυφών) και άρα πάρα πολλές πάνω από το T (περισσότερες από το 5% των κορυφών).

Παράδειγμα 2.6.1

ε=11001001000011111101101010100010001000010110100011000010001101001100011001100010100010111000, n=100

$N_1 = 46,\ N_0 = 47.5,\ d = και\ P - value = 0.168669 \geq 0.01$,

άρα η ε μπορεί να θεωρηθεί τυχαία ακολουθία.

2.7 Non-overlapping Template Matching Test

Ο έλεγχος αυτός εστιάζει στο πλήθος των εμφανίσεων προκαθορισμένων τμημάτων, μέσα στην υπό-εξέταση ακολουθία, τα οποία θεωρούνται μη-περιοδικά. Προσπαθεί δηλαδή να εντοπίσει γεννήτριες, οι οποίες παράγουν ακολουθίες στις οποίες εμφανίζεται πολλές φορές κάποιο προκαθορισμένο μη-περιοδικό μοτίβο. Αυτός ο έλεγχος, όπως και ο Overlapping Template Matching Test, χρησιμοποιεί ένα παράθυρο μεγέθους m (m bit), για να ανιχνεύσει ένα συγκεκριμένο μοτίβο μεγέθους m, δηλαδή μίας ακολουθίας μεγέθους m. Αν το ζητούμενο μοτίβο βρεθεί, τότε το παράθυρο επανατίθεται στον όρο (bit) μετά το μοτίβο που βρέθηκε και η αναζήτηση ξαναρχίζει. Σε περίπτωση που δεν βρεθεί το μοτίβο, το παράθυρο ολισθαίνει κατά έναν όρο (bit). Το τεστ χρησιμοποιεί ως κατανομή αναφοράς την κατανομή X^2.

Δεδομένα:

1. n=το μήκος της υπό εξέταση ακολουθίας.
2. $\varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n)$ = η υπό εξέταση ακολουθία.
3. m= το μήκος του κάθε μοτίβου (ακολουθίας) που αναζητούμε (ως μη-περιοδικό).
4. B= το μη περιοδικό μοτίβο μεγέθους m που αναζητούμε μέσα στην ε. Το B είναι, ουσιαστικά, μια δυαδική ακολουθία μήκους m. Ο κώδικας του τεστ παρέχει μια βιβλιοθήκη προτύπων από μη περιοδικά μοτίβα, από την οποία προκαθορίζεται το B. Τα πρότυπα που παρέχονται είναι για $m = 2, 3, \dots, 10$.
5. M= το μήκος της υπακολουθίας της ε που θα ελεγχθεί.
6. N= το πλήθος των ανεξάρτητων blocks. Το πρόγραμμα θέτει $N = 8$ (Συνιστώμενα μεγέθη: $m = 9$ ή $m = 10$, για να έχουν νόημα τα αποτελέσματα. Αν και ο κώδικας του προγράμματος θέτει $N = 8$, ο κώδικας μπορεί να μεταβληθεί για να πάρει και άλλες τιμές το N, υπό τις προϋποθέσεις: α)$N \leq 100$, για να είναι έγκυρες οι τιμές των $P - values$ και β) $M > 0.01 \cdot n$, με $N = floor(^n/_M)$).

Διαδικασία:

1. Η ακολουθία διαιρείται σε N ανεξάρτητα τμήματα (blocks) μήκους M.
2. Έστω $W_j, j = 1., 2, \dots, N$, ο αριθμός εμφανίσεων του B στο block j. Η αναζήτηση γίνεται ως εξής: δημιουργείται ένα παράθυρο μεγέθους m-bit στην ακολουθία και συγκρίνονται τα bits του παραθύρου με αυτά του προτύπου που αναζητούμε. Αν δεν ταιριάζουν το παράθυρο ολισθαίνει κατά ένα bit (προς τα δεξιά). Ενώ αν ταιριάζουν το παράθυρο ολισθαίνει (προς τα δεξιά) κατά m-bits, έτσι ώστε το επόμενο παράθυρο να ξεκινήσει με το πρώτο bit που δεν ταυτίζεται με αυτά του προτύπου.
3. $\mu = {(M - m + 1)}/{2^m}, \qquad \sigma^2 = M(\frac{1}{2^m} - \frac{2m-1}{2^{2m}})$, (κατά την υπόθεση της τυχαιότητας, υπολογίζονται τα θεωρητικά μεγέθη της μέσης τιμής μ και της διασποράς σ^2).
4. $\chi^2(obs) = \sum_{j=1}^{N} \frac{(W_j - \mu)^2}{\sigma^2}$,
   (αποτελεί ένα μέτρο για το πόσο καλά ο παρατηρούμενος αριθμός 'ταιριασμάτων' του προτύπου ταιριάζει με αυτόν που θα αναμέναμε -κατά την υπόθεση της τυχαιότητας-).
5. $P - value = igamc(\frac{N}{2}, \frac{\chi^2(obs)}{2})$,
   (θα υπολογιστούν πολλές $P - values$, θα υπολογίζεται μια $P - value$ για κάθε πρότυπο. Για $m = 9$ μπορούν να υπολογιστούν μέχρι και 148 $P - values$. Ενώ για $m = 10$ μέχρι και 284).

Συμπέρασμα: Η ε δεν είναι τυχαία και αποτυγχάνει στο τεστ αν $P - value < 0.01$. Αλλιώς (αν $P - value \geq 0.01$) η ε είναι τυχαία.

Σημείωση: Αν η τιμή της $P - value$ είναι πολύ μικρή (<0.01), τότε η ακολουθία έχει μη-κανονικές εμφανίσεις του πιθανού πρότυπου μοτίβων.

Παράδειγμα 2.7.1

Για $\varepsilon = 10100100101110010110, n = 20, N = 2$ και $M = 10$

(βήμα 1^o)

Τα δυο blocks είναι: 1010010010 και 1110010110

(βήμα 2^o)

Αν $m = 3$ και $B = 001$ έχουμε:

Θέση bit	Block1		Block2	
	Bits	W_1	Bits	W_2
1-3	101	0	111	0
2-4	010	0	110	0
3-5	100	0	100	0
4-6	001(ταιριάζει)	1	001(ταιριάζει)	1
5-7	Δεν εξετάζετε		Δεν εξετάζετε	
6-8	Δεν εξετάζετε		Δεν εξετάζετε	
7-9	001(ταιριάζει)	2	011	1
8-10	010	2	110	1

Άρα $W_1 = 2$ και $W_2 = 1$.

(βήμα 3^o)

$\mu = \frac{10-3+1}{2^3}, = 1, \sigma^2 = 10 \cdot \left(\frac{1}{2^3} - \frac{2 \cdot 3 - 1}{2^{2 \cdot 3}}\right) = 0.46875$

(βήμα 4^o)

$$\chi^2(obs) = \frac{1 + 0}{0.46875} = 2.133333$$

(βήμα 5^o)

$$P - value = igamc\left(\frac{2}{2}, \frac{2.133333}{2}\right) = 0.344154 \geq 0.01,$$

άρα η ε μπορεί να θεωρηθεί τυχαία ακολουθία.

2.8 Overlapping Template Matching Test

Ο έλεγχος αυτός, όπως και ο 2.7, εστιάζει στον αριθμό εμφανίσεων προκαθορισμένων τμημάτων-ακολουθιών. Πάλι χρησιμοποιείται ένα $m - bit$ παράθυρο, για να ανιχνευθεί ένα $m - bit$ μοτίβο. Αν το μοτίβο δεν βρεθεί έχουμε την ίδια διαδικασία με το παράθυρο να ολισθαίνει προς τα δεξιά κατά ένα bit. Η διαφορά με το 2.7 είναι ότι σε περίπτωση που ανιχνευθεί το μοτίβο, το παράθυρο θα ολισθήσει μόνο ένα bit προς τα δεξιά για να ξαναρχίσει η αναζήτηση. Το τεστ χρησιμοποιεί ως κατανομή αναφοράς την κατανομή X^2.

Δεδομένα:

1. n=το μήκος της υπό εξέταση ακολουθίας.
2. $\varepsilon = (\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_n) =$ η υπό εξέταση ακολουθία.
3. m= το μήκος του μοτίβου, σε αυτήν την περίπτωση το μήκος της ροής των άσσων.

4. B= το μοτίβο μεγέθους m που αναζητούμε μέσα στην ε.
5. K= ο αριθμός των βαθμών ελευθερίας. Το πρόγραμμα θέτει $K = 5$.
6. M= το μήκος της υπακολουθίας της ε που θα ελεγχθεί. Το πρόγραμμα θέτει $M = 1032$.
7. N= το πλήθος των ανεξάρτητων blocks του n. Το πρόγραμμα θέτει $N = 968$. (Συνιστώμενα μεγέθη: Τα K, M, N πρέπει να επιλέγονται έτσι ώστε $n \geq 10^6$. Μπορούν να επιλεγούν διάφορες τιμές για το m, αλλά η NIST προς το παρόν προτείνει $m = 9$ ή $m = 10$. Αν όμως προτιμηθούν άλλες τιμές για τις μεταβλητές, αυτό θα πρέπει να γίνει έτσι ώστε:
   - $n \geq mn$.
   - $N \cdot min\pi_i > 5$, (για την επιλογή του N).
   - $\lambda = {(M - m + 1)}/{2^m} \approx 2$.
   - $m \approx log_2 M$, (για την επιλογή του m).
   - $K \approx 2\lambda$, (για την επιλογή του K). Αν επιλεγεί $K \neq 5$, οι τιμές των π_i θα πρέπει να επαναϋπολογιστούν (βλέπε a statistical test suite, section 3.8).

Διαδικασία:

1. Η ακολουθία διαιρείται σε N ανεξάρτητα τμήματα (blocks) μήκους M.
2. Υπολογίζετε ο αριθμός εμφανίσεων του B σε κάθε ένα από τα N blocks. Η αναζήτηση γίνεται ως εξής: δημιουργείται ένα παράθυρο μεγέθους m-bit στην ακολουθία και συγκρίνονται τα bits του παραθύρου με αυτά του B και υπάρχει ένας καταμετρητής του οποίου την τιμή αυξάνουμε κάθε φορά που το μοτίβο ταιριάζει. Το παράθυρο ολισθαίνει προς τα δεξιά κατά μια θέση (bit) μετά από κάθε εξέταση, Καταγράφεται ο αριθμός εμφανίσεων του B σε κάθε block με την αύξηση ενός δείκτη $v_i, i = 0, \ldots, 5$, έτσι ώστε: το v_0 να αυξάνεται όταν δεν υπάρχουν εμφανίσεις του B στην υπακολουθία, το v_1 αυξάνεται για μια εμφάνιση του B στην υπακολουθία, ... , το v_4 αυξάνεται για 4 εμφανίσεις του B στην υπακολουθία, ενώ το v_5 αυξάνεται για 5 και πάνω εμφανίσεις του B.
3. $\lambda = {(M - m + 1)}/{2^m}$ και $\eta = {\lambda}/{2}$,
   (κανονικά χρησιμοποιούνται για να υπολογιστούν οι θεωρητικές πιθανότητες π_i που αντιστοιχούν στις κλάσεις του v_0, το πρόγραμμα όμως δουλεύει με συγκεκριμένες τιμές των π_i, βάσει των απαιτήσεων του).
4. $\chi^2(obs) = \sum_{i=0}^{5} \frac{(v_i - N\pi_i)^2}{N\pi_i}$, όπου $\pi_0 = 0.364091$, $\pi_1 = 0.185659$, $\pi_2 = 0.139381$, $\pi_3 = 0.100571$, $\pi_4 = 0.070432$, $\pi_5 = 0.139865$.

(αποτελεί ένα μέτρο για το πόσο καλά ο παρατηρούμενος αριθμός 'ταιριασμάτων' του προτύπου ταιριάζει με αυτόν που θα αναμέναμε =κατά την υπόθεση της τυχαιότητας-)

5. $P - value = igamc(\frac{5}{2}, \frac{\chi^2(obs)}{2})$

Συμπέρασμα: Η ε δεν είναι τυχαία και αποτυγχάνει στο τεστ αν $P - value < 0.01$. Αλλιώς (αν $P - value \geq 0.01$) η ε είναι τυχαία.

Σημείωση: Για το πρότυπο $B = 11$, αν σε όλο το μήκος της ακολουθίας υπήρχαν πολλές ροές άσσων μεγέθους 2-bit, τότε: α) η τιμή της v_5 θα ήταν πολύ μεγάλη, β) η τιμή της $\chi^2(obs)$ θα ήταν πολύ μεγάλη, γ) η τιμή της P-value θα ήταν μικρή (<0.01) και δ) το αποτέλεσμα θα ήταν μη-τυχαιότητα για την ακολουθία.

Παράδειγμα 2.8.1

Για ε=10111011110010110100011100101110111110000101101001, n=50, K=2, M=10 και N=5

(βήμα 1^o)

Τα 5 blocks είναι: 1011101111, 0010110100, 0111001011, 1011111000, και 0101101001

(βήμα 2^o)

Για m=2 και B=11 η διαδικασία εξέτασης του πρώτου block συνεχίζεται ως εξής:

Θέση bit	Bits	Αρ. Εμφανίσεων του B=11
1-2	10	0
2-3	01	0
3-4	11(ταιριάζει)	1
4-5	11(ταιριάζει)	2
5-6	10	2
6-7	01	2
7-8	11(ταιριάζει)	3
8-9	11(ταιριάζει)	4
9-10	11(ταιριάζει)	5

Άρα μετά την εξέταση του πρώτου block, έχουμε 5 εμφανίσεις/συμβάντα του B και επομένως: $v_0 = v_1 = v_2 = v_3 = v_4 = 0$ και $v_5 = 1$.

Όμοια εξετάζονται και τα υπόλοιπα blocks. Και έτσι το 11 εμφανίζεται: στο δεύτερο block 2 φορές, στο τρίτο block 3 φορές, στο τέταρτο block 4 φορές και στο πέμπτο block μια φορά.

Άρα $v_0 = 0,\ v_1 = v_2 = v_3 = v_4 = v_5 = 1$

(βήμα 3^o)

$\lambda = {(10 - 2 + 1)}/{2^2} = 2.25$ και $\eta = {2.25}/{2} = 1.125.$

(βήμα 4^o)

Επειδή το παράδειγμα έχει μη επιτρεπτές εισόδους για το πρόγραμμα (για λόγους μελέτης και κατανόησης του αναγνώστη), επηρεάζονται και οι τιμές των π_i οι οποίες επαναϋπολογίζονται για το συγκεκριμένο ως εξής:$\pi_0 = 0.324652,\ \pi_1 = 0.182617,\ \pi_2 = 0.142670,\ \pi_3 = 0.106645,\ \pi_4 = 0.077147,\ \pi_5 = 0.166269.$

Τότε $\chi^2(obs) = 3.167729.$

(βήμα 5^o)

$P - value = igamc(\frac{5}{2}, \frac{3.167729}{2})$=0.274932.

2.9 Maurer's "Universal Statistical" Test

Ο έλεγχος αυτός εστιάζει στο πλήθος των στοιχείων (bits) ανάμεσα σε δύο πρότυπα που ταιριάζουν (πράγμα που αποτελεί ένα μέτρο που συνδέεται με το μήκος μιας συμπιεσμένης ακολουθίας). Έχοντας ως σκοπό να ανιχνεύσει αν η ακολουθία μπορεί να συμπιεστεί σημαντικά χωρίς απώλεια πληροφοριών, ή όχι. Το τεστ χρησιμοποιεί ως κατανομή αναφοράς την ημι-κανονική κατανομή (η μία της πλευρά είναι διάφορη της κανονικής κατανομής).

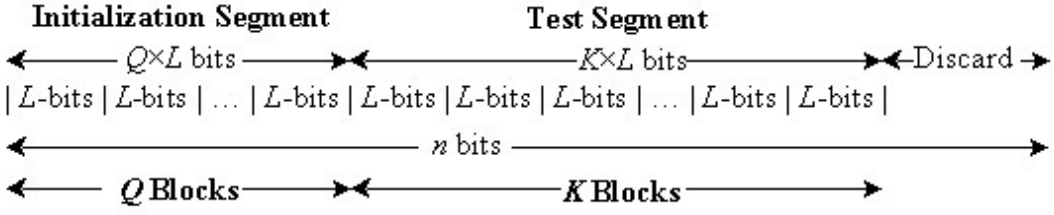
Δεδομένα:

1. n=το μήκος της υπό εξέταση ακολουθίας.
2. $\varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n) =$ η υπό εξέταση ακολουθία.
3. L= το μήκος του κάθε block. (Η χρήση της L ως μέγεθος block, δεν είναι συμβιβαστή με τον μεγέθους block συμβολισμό M, που χρησιμοποιείται στα άλλα τεστ. Παρόλα αυτά, η χρήση ως μέγεθος block καθορίστηκε στην αρχική διατύπωση του Maurer's Test).
4. Q=το πλήθος το blocks στην ακολουθία που ξεκινάει το τεστ. (Συνιστώμενα μεγέθη: Το τεστ απαιτεί μεγάλο μήκος για την ε $(n \geq (Q + K)L)$. Πρέπει: $6 \leq L \leq 16$, $Q = 2 \cdot 10^L$ και $K = floor({n}/{L}) - Q \approx 1000 \cdot 2^L$. Για αυτούς τους λόγους οι τιμές των n, L και Q πρέπει να επιλέγονται σύμφωνα με τον πίνακα:

n	L	$Q = 10 \cdot \mathbf{2^L}$
≥387,840	*6*	*640*
≥904,960	*7*	*1280*
≥2,068,480	*8*	*2560*
≥4,654,080	*9*	*5120*
≥10,342,400	*10*	*10240*
≥22,753,280	*11*	*20480*
≥49,643,520	*12*	*40960*
≥107,560,960	*13*	*81920*
≥231,669,760	*14*	*163840*
≥496,435,200	*15*	*327680*
≥1,059,061,760	*16*	*655360*

Διαδικασία:

1. Η ε χωρίζεται σε δύο τμήματα: α) το τμήμα που ξεκινά το τεστ (initialization segment), που αποτελείται από Q το πλήθος μη αλληλεπικαλυπτόμενα (non-overlapping) blocks μεγέθους L, και β) από το τμήμα ελέγχου (test seqment), που αποτελείται από τα υπόλοιπα K το πλήθος μη αλληλεπικαλυπτόμενα blocks μεγέθους L, όπου $K = floor(^{n}/_{L}) - Q$. Αν υπάρχουν στοιχεία (bits) στο τέλος της ακολουθίας ε, τα οποία δεν επαρκούν να σχηματίσουν block μεγέθους L, τότε αυτά απορρίπτονται.



2. Χρησιμοποιώντας το τμήμα που ξεκινά το τεστ, δημιουργείται ένας πίνακας για κάθε πιθανή τιμή μεγέθους L (η οποία χρησιμοποιείται ως δείκτης στον πίνακα), ο αριθμός της τελεταίας εμφάνισης του τελευταίου block μεγέθους L σημειώνεται στον πίνακα (δηλαδή $T_j = i$, για $i = 1, \dots, Q$, όπου j η δεκαδική αναπαράσταση των περιεχομένων του i-στου block μεγέθους L.
3. Εξετάζετε κάθε ένα από τα K blocks του τμήματος ελέγχου και προσδιορίζεται ο αριθμός των blocks από την τελευταία εμφάνιση του ίδιου block μεγέθους L (δηλαδή $i - T_j$). Προστίθεται η υπολογισμένη απόσταση μεταξύ των επανεμφανίσεων του ίδιου block μεγέθους L σε έναν συσωρευμένο log_2

άθροισμα όλων των διαφορών που ανιχνεύθηκαν στα K blocks (δηλαδη $sum = sum + log_2(i - T_j)$).

4. $f_n = \frac{1}{K}\sum_{i=Q+1}^{Q+K} log_2(i - T_j)$, όπου T_j είναι η είσοδος του πίνακα που αντιστοιχεί στη δεκαδική αναπαράσταση του i-στου block μεγέθους L.
   (Η f_n είναι το άθροισμα των log_2 αποστάσεων ανάμεσα σε πρότυπα μεγέθους L που ταιριάζουν. Αποτελεί, δηλαδή, το άθροισμα των ψηφίων που βρίσκονται ανάμεσα σε πρότυπα μεγέθους L που ταιριάζουν.)
5. $P - value = erfc(\left|\frac{f_n - expectedValue(L)}{\sqrt{2}c}\right|)$, όπου $\sigma = c\sqrt{\frac{varience(L)}{K}}$ με $c = 0.7 - \frac{0.8}{L} + (4 + \frac{32}{L})\frac{K^{-3/L}}{15}$ και

L	expextedValue	variance
6	5.2177052	2.954
7	6.1962507	3.125
8	7.1836656	3.238
9	8.1764248	3.311
10	9.1723243	3.356
11	10.170032	3.384
12	11.168765	3.401
13	12.168070	3.410
14	13.167693	3.416
15	14.167488	3.419
16	15.167379	3.421

(Κατά την υπόθεση της τυχαιότητας, ο δειγματικός μέσος, expectedValue(L), είναι η θεωρητικώς αναμενόμενη τιμή που θα περιμέναμε στη θέση της f_n για το δοσμένο μήκος L. Και σ η θεωρητική τυπική απόκλιση.)

Συμπέρασμα: Η ε δεν είναι τυχαία και αποτυγχάνει στο τεστ αν $P - value < 0.01$. Αλλιώς (αν $P - value \geq 0.01$) η ε είναι τυχαία.

Σημείωση: Το τεστ αποτυγχάνει, όταν η τιμή της f_n διαφέρει σημαντικά από τη θεωρητικώς αναμενόμενη expectedValue, όταν δηλαδή η ακολουθία μπορεί να συμπιεστεί σημαντικά.

Παράδειγμα 2.9.1

Για ε= $0101101001110101011 1, n = 20, L = 2$ και $Q = 4$.

(βήμα $1^{ο}$)

Τότε K=6. το τμήμα που ξεκινά το τεστ (initialization segment) είναι: 01011010 και το τμήμα ελέγχου (test segment) είναι: 011101010111. Τα L-bit blocks παρουσιάζονται στον κάτωθι πίνακα:

Block	Τύπος	Περιεχομενο
1	initialization segment	01
2		01
3		10
4		10
5	test segment	01
6		11
7		01
8		01
9		01
10		11

(βήμα 2^o)

Χρησιμοποιώντας τα 4 initialization blocks έχουμε τον πίνακα που θα μας βοηθήσει για τον υπολογισμό των πρώτων T_i που θα χρησιμοποιηθούν.

	Πιθανές L-bit τιμές			
	00 (αποθηκεύεται στο $\boldsymbol{T_0}$)	01 (αποθηκεύεται στο $\boldsymbol{T_1}$)	10 (αποθηκεύεται στο $\boldsymbol{T_2}$)	11 (αποθηκεύεται στο $\boldsymbol{T_3}$)
Initialization	0	2	4	0

(βήμα 3^o)

Για το block 5 (το 1^o που εξετάζεται): το 5 είναι τοποθετημένο στην ¨01¨ σειρά του πίνακα (άρα , T_1), και τότε $sum = log_2(5-2) = 1.584962501$.

Για το block 6: το 6 είναι τοποθετημένο στην ¨11¨ σειρά του πίνακα (άρα , T_3), $sum = 1.584962501 + log_2(6-0) = 4.169925002$.

Για το block 7: το 7 είναι τοποθετημένο στην ¨01¨ σειρά του πίνακα με προηγούμενη εμφάνιση στην $5^η$ σειρά, $sum = 4.169925002 + log_2(7-5) = 5.169925002$.

Για το block 8: το 8 είναι τοποθετημένο στην ¨01¨ σειρά του πίνακα με προηγούμενη εμφάνιση στην $7^η$ σειρά, $sum = 5.169925002 + log_2(8-7) = 5.169925002$.

Για το block 9: το 9 είναι τοποθετημένο στην ¨01¨ σειρά του πίνακα με προηγούμενη εμφάνιση στην $8^η$ σειρά, $sum = 5.169925002 + log_2(9-8) = 5.169925002$.

Για το block 10: το 10 είναι τοποθετημένο στην ¨11¨ σειρά του πίνακα με προηγούμενη εμφάνιση στην $6^η$ σειρά, $sum = 5.169925002 + log_2(10-6) = 7.169925002$ . .

(βήμα 4^o)

$f_n = \frac{7.169925002}{6} = 1.1949875$.

(βήμα 5^o)

Επειδή το τεστ δεν ενδείκνυται για L=2, δεν υπάρχει αντίστοιχη expectedValue στον πίνακα. Προτιμήθηκε όμως για το παράδειγμα για λόγους διευκόλυνσης του αναγνώστη, και επομένως χρειάστηκε η εισαγωγή της expectedValue για L=2 αποκλειστικά και μόνο για το παρόν παράδειγμα.

$$P - value = erfc\left(\left|\frac{1.1949875 - 1.5374383}{\sqrt{2}\sqrt{1.338}}\right|\right) = 0.767189$$

2.10 Linear Complexity Test

Ο έλεγχος αυτός εστιάζει στο μήκος ενός καταχωρητή ολίσθησης με γραμμική ανάδραση LFSR (παρουσιάζονται αναλυτικά στο Κεφάλαιο 3). Σκοπός του είναι να αποφανθεί αν η ακολουθία είναι αρκετά σύνθετη για να θεωρηθεί τυχαία. Τυχαίες ακολουθίες χαρακτηρίζονται από μεγαλύτερους LFSR's. Επομένως αν έχουμε πολύ μικρό καταχωρητή συνεπάγεται μη τυχαιότητα. Το τεστ χρησιμοποιεί ως κατανομή αναφοράς την κατανομή X^2.

Δεδομένα:

1. n=το μήκος της υπό εξέταση ακολουθίας.
2. $\varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n) =$ η υπό εξέταση ακολουθία.
3. M= το μήκος ενός block.
4. K= ο αριθμός των βαθμών ελευθερίας. Ο κώδικας του προγράμματος έχει γραφεί με K=6.
   (Συνιστώμενα μεγέθη: $n \geq 10^6$.Επιπλέον $500 \leq M \leq 5000$ και $N \geq 200$, για να είναι έγκυρο το τεστ.)

Διαδικασία:

1. Χωρίζεται η ε σε N το πλήθος ανεξάρτητα blocks μεγέθους M, όπου $n = MN$.
2. Για $i = 1, \dots, N$ καθορίζεται η γραμμική πολυπλοκότητα L_i για κάθε ένα από τα N blocks με τη χρήση του γνωστού αλγορίθμου Berlekamp-Massey[3]. L_i είναι το μήκος της μικρότερης (υπ)ακολουθίας που παράγει όλα τα στοιχεία (bits) στο block i. Για κάθε υπακολουθία μεγέθους L_i, χρησιμοποιούμε το $mod2$ άθροισμα

[3] Παρουσιάζετε στο Παράρτημα Α, εναλλακτικά
The Handbook of Applied Cryptography; A. Menezes, P. Van Oorschot and S. Vanstone; CRC Press, 1997

κάποιων στοιχείων της, για να πάρουμε το αποτέλεσμα αυτού του αθροίσματος ως το επόμενο στοιχείο (το $L_i + 1$ bit) της ακολουθίας.

3. $\mu = \frac{M}{2} + \frac{9+(-1)^{M+1}}{36} - \frac{\frac{M}{3}+\frac{2}{9}}{2^M}$,
   (αποτελεί τη θεωρητική μέση τιμή κατά την υπόθεση της τυχαιότητας)
4. $T_i = (-1)^M \cdot (L_\iota - M) + \frac{2}{9}$, για $i = 1, \dots, N$
5. $(v_0, v_1, \dots, v_6) = (0,0, \dots 0)$
   Για $i = 1, \dots, N$
   Αν $T_i \leq -2.5$ τότε $v_0 = v_0 + 1$
   Αλλιώς αν $-2.5 < T_i \leq -1.5$, τότε $v_1 = v_1 + 1$
   Αλλιώς αν $-1.5 < T_i \leq -0.5$, τότε $v_2 = v_2 + 1$
   Αλλιώς αν $-0.5 < T_i \leq 0.5$, τότε $v_3 = v_3 + 1$
   Αλλιώς αν $0.5 < T_i \leq 1.5$, τότε $v_4 = v_4 + 1$
   Αλλιώς αν $1.5 < T_i \leq 2.5$, τότε $v_5 = v_5 + 1$
   Αλλιώς αν $T_i > 2.5$, τότε $v_6 = v_6 + 1$
   (Αποτελεί έναν τρόπο καταγραφής των τιμών των T_i στις $v_0, v_1, \dots, v_6$.)
6. $\chi^2(obs) = \sum_{i=0}^{K} \frac{(v_i - N\pi_i)^2}{N\pi_i}$, όπου $\pi_0 = 0.010417$, $\pi_1 = 0.03125$, $\pi_2 = 0.125$, $\pi_3 = 0.5$, $\pi_4 = 0.25$, $\pi_5 = 0.0625$, $\pi_6 = 0.02083$
7. $P - value = igamc(\frac{K}{2}, \frac{\chi^2(obs)}{2})$

Συμπέρασμα: Η ε δεν είναι τυχαία και αποτυγχάνει στο τεστ αν $P - value < 0.01$. Αλλιώς (αν $P - value \geq 0.01$) η ε είναι τυχαία.

Σημείωση: Η αποτυχία του τεστ ($P - value < 0.01$) πηγάζει από το ότι οι παρατηρούμενοι μετρητές των συχνοτήτων των T_i (δηλαδή τα $v_j, j = 0, \dots, 6$) διαφέρουν από τις αναμενόμενες τιμές. Αφού θα έπρεπε η διαταραχή των συχνοτήτων των T_i (δηλαδή των v_j) να είναι ανάλογη των υπολογισμένων π_j.

Παράδειγμα 2.10.1

Θα δοθούν για λόγους κατανόησης και ανάγνωσης τα βήματα της διαδικασίας που μπορούν να αναλυθούν πρακτικά σε γραπτό κείμενο.

(βήματα 2^o)

Έστω ότι M=13 και το block που θα εξεταστεί είναι το 1101011110001 Τότε $L_i = 4$ και η ακολουθία παράγεται από τη mod2 πρόσθεση του $1^{ου}$ και του $4^{ου}$ στοιχείου (bit) μέσα σε μία υπακολουθία μεγέθους 4 (bit) για την παραγωγή του $5^{ου}$ στοιχείου (bit). Η διερεύνηση συνεχίζεται ως κάτωθι:

	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5
Τα πρώτα 4 bits και το αποτέλεσμα 5^o bit	1	1	0	1	0
Bits 2-5 και το αποτέλεσμα 6^o bit	1	0	1	0	1
Bits 3-6 και το αποτέλεσμα 7^o bit	0	1	0	1	1
Bits 4-7 και το αποτέλεσμα 8^o bit	1	0	1	1	1
Bits 5-8 και το αποτέλεσμα 9^o bit	0	1	1	1	1
Bits 6-9 και το αποτέλεσμα 10^o bit	1	1	1	1	0
Bits 7-10 και το αποτέλεσμα 11^o bit	1	1	1	0	0
Bits 8-11 και το αποτέλεσμα 12^o bit	1	1	0	0	0
Bits 9-12 και το αποτέλεσμα 13^o bit	1	0	0	0	1

Αυτός ο LFSR λειτουργεί, αν δεν λειτουργούσε θα δοκιμαζόταν άλλη (γραμμική) συνάρτηση (πχ άθροισμα mod2 του $1^{ου}$, του $2^{ου}$ και του $4^{ου}$ στοιχείου και τα αποτέλεσμα να είναι το 5^o στοιχείο).

(βήμα 3^o)

$\mu = \frac{13}{2} + \frac{9+(-1)^{13+11}}{36} - \frac{\frac{13}{3}+\frac{2}{9}}{2^{13}} = 6.777222$.

(βήμα 4^o)

$T_i = (-1)^{13} \cdot (4 - 13) + \frac{2}{9} = 2.999444$

2.11 Serial Test

Ο έλεγχος αυτός εστιάζει στη συχνότητα όλων των πιθανών αλληλεπικαλυπτόμενων μοτίβων μεγέθους m (m-bit) κατά μήκος ολόκληρης της ακολουθίας. Αποσκοπώντας να καθορίσει αν το πλήθος των εμφανίσεων των 2^m αλληλεπικαλυπτόμενων μοτίβων μεγέθους m, είναι αυτός που θα αναμέναμε από μια τυχαία ακολουθία. Στις τυχαίες ακολουθίες κάθε m-bit μοτίβο έχει την ίδια πιθανότητα εμφάνισης με οποιοδήποτε άλλο, γεγονός που δίνει μια ομοιομορφία σε αυτές (τις ακολουθίες). Το τεστ είναι ισοδύναμο με το Frequency Test για m=1. Το τεστ χρησιμοποιεί ως κατανομή αναφοράς την κατανομή X^2.

Δεδομένα:

1. n=το μήκος της υπό εξέταση ακολουθίας.
2. $\varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n) =$ η υπό εξέταση ακολουθία.
3. m= το μήκος του κάθε block.
   (Συνιστώμενα μεγέθη: τα m, n πρέπει να επιλεγούν έτσι ώστε $m < floor(log_2 n) - 2$.)

Διαδικασία:

1. Σχηματίζεται μια επαυξημένη ακολουθία ε′. Η επέκταση γίνεται αν στο τέλος της ε προσαρτήσουμε τα πρώτα m-1 στοιχεία της.
2. Καθορίζονται οι συχνότητες όλων των πιθανών αλληλεπικαλυπτόμενων block μεγέθους: m, m-1 και m-2. Έστω: $v_{i_1 \dots i_m}$ η συχνότητα του μοτίβου $i_1 \dots i_m$, $v_{i_1 \dots i_{m-1}}$ η συχνότητα του μοτίβου $i_1 \dots i_{m-1}$ και $v_{i_1 \dots i_{m-2}}$ η συχνότητα του μοτίβου $i_1 \dots i_{m-2}$.
3. $\psi_m^2 = \frac{2^m}{n}\sum_{i_1 \dots i_m}(v_{i_1 \dots i_m} - \frac{n}{2^m})^2 = \frac{2^m}{n}\sum_{i_1 \dots i_m}(v_{i_1 \dots i_m})^2 - n$

   $\psi_{m-1}^2 = \frac{2^{m-1}}{n}\sum_{i_1 \dots i_{m-1}}(v_{i_1 \dots i_{m-1}} - \frac{n}{2^{m-1}})^2 = \frac{2^{m-1}}{n}\sum_{i_1 \dots i_{m-1}}(v_{i_1 \dots i_{m-1}})^2 - n$

   $\psi_{m-2}^2 = \frac{2^{m-2}}{n}\sum_{i_1 \dots i_{m-2}}(v_{i_1 \dots i_{m-2}} - \frac{n}{2^{m-2}})^2 = \frac{2^{m-2}}{n}\sum_{i_1 \dots i_{m-2}}(v_{i_1 \dots i_{m-2}})^2 - n$
4. $\nabla\psi_m^2 = \psi_m^2 - \psi_{m-1}^2$ και $\nabla^2\psi_m^2 = \psi_m^2 - 2\psi_{m-1}^2 + \psi_{m-2}^2$, όπου $\psi_0^2 = \psi_{-1}^2 = 0$. (Τα $\nabla\psi_m^2(obs)$ και $\nabla^2\psi_m^2(obs)$ αποτελούν ένα μέτρο για το πόσο καλά οι παρατηρούμενες συχνότητες των μοτίβων μεγέθους m ταιριάζουν με αυτές που θα αναμέναμε θεωρητικά.)
5. $P - value1 = igamc(2^{m-2}, \nabla\psi_m^2)$ και $P - value2 = igamc(2^{m-3}, \nabla\psi_m^2)$

Συμπέρασμα: Η ε δεν είναι τυχαία και αποτυγχάνει στο τεστ αν $P - value < 0.01$. Αλλιώς (αν $P - value \geq 0.01$) η ε είναι τυχαία.

Σημείωση: Από τη στιγμή που $P - value1 \geq 0.01$ και $P - value2 \geq 0.01$ συμπαιρένουμε ότι η ακολουθία είναι τυχαία. Αν η τιμή της $\nabla^2\psi_m^2$ή της $\nabla\psi_m^2$είναι πολύ μεγάλη, αυτό σημαίνει ότι δεν έχουμε ομοιομορφία στο block μεγέθους m.

Παράδειγμα 2.11.1

Για n=10 και ε=0011011101.

(βήμα 1)

Για m=3, τότε ε′=001101110100, για m=2 τότε ε′= 00110111010, για m=1 τότε ε′=0011011101

(βήμα 2)

Για m=3, m-1=2 και m-2=1.

Οι συχνότητες όλων των 3-bit blocks είναι: $v_{000} = 0$, $v_{001} = 1$, $v_{010} = 1$, $v_{011} = 2$, $v_{100} = 1$, $v_{101} = 2$, $v_{110} = 2$, $v_{111} = 0$

Οι συχνότητες όλων δυνατών των 2-bit blocks είναι: $v_{00} = 1$, $v_{01} = 3$, $v_{10} = 3$, $v_{11} = 3$

Οι συχνότητες όλων δυνατών των 1-bit blocks είναι: $v_0 = 4$, $v_1 = 6$

(βήμα 3)

$\psi_3^2 = \frac{2^3}{10}(0 + 1 + 1 + 4 + 1 + 4 + 4 + 1) - 10 = 2.8$

$\psi_2^2 = \frac{2^2}{10}(1 + 9 + 9 + 9) - 10 = 1.2$

$\psi_1^2 = \frac{2}{10}(16 + 36) - 10 = 0.4$

(βήμα 4)

$\nabla\psi_m^2 = \psi_m^2 - \psi_{m-1}^2 = \psi_3^2 - \psi_2^2 = 2.8 - 1.2 = 1.6$

$\nabla^2\psi_3^2 = \psi_3^2 - 2\psi_2^2 + \psi_1^2 = 0.8$

(βήμα 5)

$P - value1 = igamc(2, \frac{1.6}{2}) = 0.9057(\geq 0.01)$ και

$P - value2 = igamc(1, \frac{0.8}{2}) = 0.8805(\geq 0.01)$

Άρα η ε μπορεί να θεωρηθεί τυχαία ακολουθία.

2.12 Approximate Entropy Test

Ο έλεγχος αυτός, όπως και ο 2.11, εστιάζει στις συχνότητες όλων των πιθανών αλληλεπικαλυπτόμενων μοτίβων μεγέθους m κατά μήκος ολόκληρης της ακολουθίας. Σκοπός του είναι να συγκρίνει τις συχνότητες αλληλεπικαλυπτόμενων blocks τα οποία έχουν διαδοχικά/παρακείμενα μήκη (αν, δηλαδή, το ένα έχει μήκος m το άλλο θα έχει m+1), με αντίστοιχα αποτελέσματα που θα αναμέναμε από μια τυχαία ακολουθία. Το τεστ χρησιμοποιεί ως κατανομή αναφοράς την κατανομή X^2.

Δεδομένα:

1. n=το μήκος της υπό εξέταση ακολουθίας.
2. $\varepsilon = (\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_n) =$ η υπό εξέταση ακολουθία.
3. m= το μήκος του κάθε block.(Στο τεστ θα είναι το πρώτο μέγεθος block που θα χρησιμοποιείται και m+1 το δεύτερο.)
   (Συνιστώμενα μεγέθη: τα m, n πρέπει να επιλεγούν έτσι ώστε $m < floor(log_2 n) - 5$.)

Διαδικασία:

1. Σχηματίζεται μια επαυξημένη ακολουθία ε΄, έτσι ώστε να δημιουργηθούν n αλληλεπικαλυπτόμενες ακολουθίες μεγέθους m, για να επιτευχθεί αυτό στο τέλος της ε προσαρτόνται τα πρώτα m-1 στοιχεία της.
2. Δημιουργείται ένας μετρητής συχνοτήτων για τα n αλληλεπικαλυπτόμενα blocks (πχ αν σε ένα block περιέχονται τα στοιχεία ε_j μέχρι το ε_{j+m-1} τότε αυτό εξετάζεται τη χρονική στιγμή j,αντίστοιχα ένα block που περιέχει στοιχεία από το ε_{j+1} μέχρι το ε_{j+m}αυτό εξετάζεται τη χρονική στιγμή j+1). Έστω ότι ο

μετρητής των πιθανών τιμών μεγέθους m (m+1) συμβολίζεται ως C_i^m, όπου i είναι μια τιμή μεγέθους m.

3. $C_i^m = \frac{\#i}{n}, \forall i$ (με #i συμβολίζουμε το πλήθος των i, δηλαδή τον αριθμό εμφανίσεων του i).
4. $\varphi^{(m)} = \sum_{i=0}^{2^m-1} \pi_i log \pi_i$, όπου $\pi_i = C_j^3$ και $j = \log_2 i$.
5. Θέτουμε $m := m + 1$ και επαναλαμβάνονται τα βήματα 1 έως 4.
6. $X^2 = 2n\{log2 - ApEn(m)\}$, όπου $ApEn(m) = \varphi^{(m)} - \varphi^{(m+1)}$. (η X^2 αποτελεί ένα μέτρο για το πόσο καλά η παρατηρούμενη τιμή της $ApEn(m)$ ταιριάζουν με αυτές που θα περιμέναμε).
7. $P - value = igamc(2^{m-1}, \frac{X^2}{2})$

Συμπέρασμα: Η ε δεν είναι τυχαία και αποτυγχάνει στο τεστ αν $P - value < 0.01$. Αλλιώς (αν $P - value \geq 0.01$) η ε είναι τυχαία.

Σημείωση: Μικρές τιμές της ApEn(m) θα σημαίνουν σημαντική κανονικότητα, ενώ μεγάλες τιμές θα έδειχναν ουσιαστική διακύμανση ή ανωμαλία.

Παράδειγμα 2.12.1

Για ε=0100110101, n=10 και m=3

(βήμα 1)

m-1=2, ε′=010011010101 (αυτό γίνεται για κάθε τιμή του m)

(βήμα 2)

Τα blocks μεγέθους 3 είναι: 010, 100, 001, 011, 110, 101, 010, 101, 010, και 101.

Οι υπολογιζόμενοι μετρητές για τα $2^m = 2^3 = 8$ πιθανές σειρές μήκους 3 είναι: #000=0, #001=1, #010=3, #011=1, #100=1, #101=3, #110=1, #111=0

(βήμα 3)

$C_{000}^3 = 0$, $C_{001}^3 = 0.1$, $C_{010}^3 = 0.3$, $C_{011}^3 = 0.1$, $C_{100}^3 = 0.1$, $C_{101}^3 = 0.3$, $C_{110}^3 = 0.1$, $C_{111}^3 = 0$.

(βήμα 4)

$\varphi^{(3)} = 0(\log 0) + 0.1(\log 0.1) + 0.3(\log 0.3) + 0.1(\log 0.1) + 0.1(\log 0.1) + 0.3(\log 0.3) + 0.1(\log 0.1) + 0(\log 0) = -1.64341772.$

(βήμα 5)

m=4

1. ε′=0100110101010.
2. Τα blocks μεγέθους 4 είναι: 0100, 1001, 0011, 0110, 1101, 1010, 0101, 1010, 0101,1010.

   #0011 = 1, #0100 = 1, #0101 = 2, #0110 = 1, #1001 = 1, #1010 = 3,

#1101 = 1, υπάρχουν άλλα 9 πιθανά blocks που δεν εμφανίζονται καθόλου για αυτό δεν τα αναφέρουμε (πχ #0001=#1111=0)

3. $C_{0011}^4 = C_{0100}^4 = C_{0110}^4 = C_{1001}^4 = C_{1101}^4 = 0.1, C_{0101}^4 = 0.2, C_{1010}^4 = 3$, και όλες οι άλλες τιμές είναι 0.
4. $\varphi^{(4)} = 0 + 0 + 0 + 0.1(\log 0.01) + 0.1(\log 0.01) + 0.2(\log 0.02) + 0.1(\log 0.01) + 0 + 0 + 0.1(\log 0.01) + 0.3(\log 0.03) + 0 + 0 + 0.1(\log 0.01) + 0\ 0) = -1.83437197.$

   (βήμα 6)

   $ApEn(3) = -1.643418 - (-1.834372) = 0.190954$ και $X^2 = 0.502193$

   (βήμα 7)

   $P - value = igamc(2^2, \frac{0.502193}{2}) = 0.261961(\geq 0.01)$.

   Άρα η ε μπορεί να θεωρηθεί τυχαία.

Παράδειγμα 2.12.2

Το παρόν αποτελεί ένα παράδειγμα μιας ακολουθίας που θα είχε νόημα να ελεγχθεί ως προς την τυχαιότητα της, την οποία το πρόγραμμα θα προσπέλαζε ως εξής: Για

ε=11001001000011111101101010100010001000010110100011000010001101001100010011000100101000101110000 n=100, m=2

$ApEn(m) = 0.665393$, $X^2 = 5.550792$, $P - value = 0.235301(\geq 0.01)$.

Άρα η ε μπορεί να θεωρηθεί τυχαία.

2.13 Cumulative Sums (Cusum) Test

Ο έλεγχος αυτός εξετάζει το μέγιστο μονοπάτι από το 0 ενός τυχαίου περιπάτου, το οποίο καθορίζεται από το άθροισμα των προσαρμοσμένων (τα 0 πάλι αντικαθιστώνται από -1) στοιχείων της ακολουθίας. Θέλοντας να καθορίσει αν το άθροισμα των υπακολουθιών της υπό εξέταση ακολουθίας, σε σχέση με αυτό που θα περιμέναμε από μια τυχαία ακολουθία, είναι αρκετά μεγάλο ή αρκετά μικρό. Αν η ακολουθία είναι τυχαία θα πρέπει το μονοπάτι του τυχαίου περιπάτου να είναι κοντά στο 0. Για ορισμένους τύπους μη-τυχαίων ακολουθιών , αυτού του είδους τα μονοπάτια είναι μεγάλα. Το τεστ χρησιμοποιεί ως κατανομή αναφοράς την κανονική κατανομή.

Δεδομένα:

1. n=το μήκος της υπό εξέταση ακολουθίας.
2. $\varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n) =$ η υπό εξέταση ακολουθία.

3. $mode$= είναι ο δείκτης βάσει του οποίου εφαρμόζουμε το τεστ, είτε από την αρχή προς το τέλος της ε (για $mode = 0$), είτε από το τέλος προς την αρχή (για $mode = 1$).
(Συνιστώμενα μεγέθη: $n \geq 100$.)

Διαδικασία:

1. χ_i=2ε_i-1, $i = 1, \dots, n$
(Μετατρέπονται τα στοιχεία της ακολουθίας ε που είναι ίσα με 0 σε -1, ενώ αυτά που είναι ίσα με 1 παραμένουν ως έχουν και παράγεται έτσι μια νέα ακολουθία $(\chi_n)_{n=1}^{\infty}$=X.
2. Αν $mode = 0$ $S_\kappa = \chi_1 + \chi_2 + \cdots + \chi_\kappa = S_{\kappa-1} + \chi_\kappa$, αλλιώς αν $mode = 1$ $S_\kappa = \chi_n + \chi_{n-1} + \cdots + \chi_{n-\kappa+1} = S_{\kappa-1} + \chi_{\nu-\kappa+1}$, για $\kappa = 1,2, \dots, n$. Δηλαδή:

$\boldsymbol{mode = 0}$	$\boldsymbol{mode = 1}$
$S_1 = \chi_1$	$S_1 = \chi_n$
$S_2 = \chi_1 + \chi_2$	$S_2 = \chi_n + \chi_{n-1}$
⋮	⋮
$S_n = \chi_1 + \chi_2 + \cdots + \chi_n$	$S_n = \chi_n + \chi_{n-1} + \cdots + \chi_1$

(Υπολογίζονται τα (μερικά) αθροίσματα $S_i, i = 1, \dots, n$ των διαδοχικά αυξανόμενων υπακολουθιών ξεκινώντας είτε από το χ_1 αν mode=0, είτε από το χ_n αν mode=1.)
3. z= $max_{1\leq\kappa\leq n}|S_\kappa|$ (εκφράζει τον μεγαλύτερο περίπατο).
4. $P - value$=1 – $\sum_{\kappa=(\frac{-n}{z}+1)/4}^{(\frac{n}{z}-1)/4}[\Phi(\frac{(4\kappa+1)z}{\sqrt{n}}) - \Phi(\frac{(4\kappa-1)z}{\sqrt{n}})] + \sum_{\kappa=(\frac{-n}{z}-3)/4}^{(\frac{n}{z}-1)/4}[\Phi(\frac{(4\kappa+3)z}{\sqrt{n}}) - \Phi(\frac{(4\kappa+1)z}{\sqrt{n}})]$,
όπου $\Phi(z)$ η συνάρτηση κατανομής της τυποποιημένης κανονικής κατανομής που ορίζεται ως $\Phi(z) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{z} e^{-u^2/2}du$

Συμπέρασμα: Η ε δεν είναι τυχαία και αποτυγχάνει στο τεστ αν $P - value < 0.01$. Αλλιώς (αν $P - value \geq 0.01$) η ε είναι τυχαία.

Σημείωση: Αν έχουμε μεγάλες τιμές για το z σημαίνει ότι: α) για $mode = 0$ στα αρχικά τμήματα της ε υπάρχουν είτε πολλά 1, είτε πολλά 0, β)για $mode = 1$ στα τελικά τμήματα της ε υπάρχουν είτε πολλά 1, είτε πολλά 0. Ενώ αν έχουμε μικρές τιμές του z σημαίνει ότι τα 0 και 1 αναμειγνύονται ομοιόμορφα κατά μήκος της ε.

Παράδειγμα 2.13.1

Για ε= 1011010111

(βήμα 1)

X=1,-1,1,1,-1,1,-1,1,1

(βήμα 2)

S_1=1, S_2=1+(-1)=0, S_3=1+(-1)+1=1, S_4=1+(-1)+1+1=2, S_5=1+(-1)+1+1+(-1)=1,

S_6 = 1+(-1)+1+1+(-1)+1=2, S_7=1+(-1)+1+1+(-1)+1+(-1)=1,

S_8=1+(-1)+1+1+(-1)+1+(-1)+1=2,

S_9=1+(-1)+1+1+(-1)+1+(-1)+1+1=3, S_{10}=1+(-1)+1+1+(-1)+1+(-1)+1+1+1=4

(βήμα 3)

P-value=0.4116588

Παράδειγμα 2.13.2

Το παρόν αποτελεί ένα παράδειγμα μιας ακολουθίας που θα είχε νόημα να ελεγχθεί ως προς την τυχαιότητα της, την οποία το πρόγραμμα θα προσπέλαζε ως εξής: Για

ε=11001001000011111101101010100010001000010110100011000010001101001100010011000100101000101110000 , n=100

Αν mode=0 z=1.6 $P-value = 0.219194 (\geq 0.01)$.

Αν mode=1 z=1.9 $P-value = 0.114866 (\geq 0.01)$.

Άρα η ε μπορεί να θεωρηθεί τυχαία.

2.14 Random Excursions Test

Ο έλεγχος αυτός εστιάζει στον αριθμό των κύκλων (διαδρομών) που έχουν ακριβώς K επισκέψεις σε έναν ¨αθροιστικό¨ τυχαίο περίπατο. Ο ¨αθροιστικός¨ τυχαίος περίπατος παράγετε από τα μερικά αθροίσματα της υπό εξέταση ακολουθίας, αν πρώτα μετατρέψουμε τα στοιχεία της που είναι ίσα με 0 σε -1 (όπως και στο 2.13). Ένας κύκλος ενός τυχαίου περιπάτου αποτελείται από μια ακολουθία βημάτων -μοναδιαίου μήκους- τα οποία είναι τυχαία και καταλήγουν εκεί όπου ξεκινούν (για το τεστ σε ένα 0). Σκοπός του τεστ είναι να καθοριστεί αν ο αριθμός των επισκέψεων σε μια συγκεκριμένη κατάσταση μέσα σε έναν κύκλο συγκλίνει ή αποκλίνει από αυτόν που θα περιμέναμε για μια τυχαία ακολουθία. Ουσιαστικά αυτό το τεστ είναι μια σειρά από 8 τεστ (και άρα 8 συμπερασμάτων), όπου έχουμε ένα τεστ και ένα συμπέρασμα για κάθε μια από τις καταστάσεις: -4, -3, -2, -1 και +1, +2, +3, +4. Το τεστ χρησιμοποιεί ως κατανομή αναφοράς την κατανομή X^2.

Δεδομένα:

1. n=το μήκος της υπό εξέταση ακολουθίας.

2. $\varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n) =$ η υπό εξέταση ακολουθία.
   (Συνιστώμενα μεγέθη: $n \geq 10^6$)

Διαδικασία:

1. χ_i=2ε_i-1, $i = 1, \dots, n$
   (Μετατρέπονται τα στοιχεία της ακολουθίας ε που είναι ίσα με 0 σε -1, ενώ αυτά που είναι ίσα με 1 παραμένουν ως έχουν και παράγεται έτσι μια νέα ακολουθία $(\chi_n)_{n=1}^{\infty}$=X).
2. $S_\kappa = \chi_1 + \chi_2 + \cdots + \chi_\kappa, \kappa = 1,2, \dots, n$
   (υπολογίζονται τα μερικά αθροίσματα των διαδοχικά αυξανόμενων υπακολουθιών της X, δηλαδή $S_1 = \chi_1$, ..., $S_n = \chi_1 + \chi_2 + \cdots \chi_n$).
3. $S = \{S_1, \dots, S_n\}$, (η ακολουθία που έχει ως όρους τα μερικά αθροίσματα). $S' = \{0, S_1, \dots, S_n, 0\}$, (προσθέτουμε στην S από ένα 0 στην αρχή και στο τέλος).
4. Έστω J το πλήθος των στοιχείων της S′ που είναι ίσα με 0, χωρίς να προσμετράτε σε αυτά το πρώτο 0 της S′. Το J είναι επιπλέον και το πλήθος των κύκλων στην S′ ,όπου ως κύκλος στην S′ θεωρείται μια υπακολουθίας της η οποία ξεκινάει από την εμφάνιση ενός στοιχείου που είναι ίσο με 0 (το πρώτο στοιχείο της υπακολουθίας), τα επόμενα στοιχεία της (εξαιρουμένου του τελευταίου) είναι διάφορα του μηδενός, και τελειώνει (έχει ως τελευταιο στοιχείο) το πρώτο στοιχείο που θα συναντήσει το οποίο έχει τιμή ίση με 0 (εκτός του πρώτου στοιχείου της υπακολουθίας). Το τελευταίο 0 ενός κύκλου μπορεί να θεωρηθεί ως πρώτο στοιχείο για έναν επόμενο κύκλο. Το πλήθος των κύκλων στην S′ είναι το πλήθος των μηδενικών διελεύσεων. Αν J<500 το τεστ δεν έχει νόημα να συνεχιστεί.
5. Για κάθε κύκλο και για κάθε κατάσταση χ ,όπου $\chi \in \mathbb{Z}$, $-4 \leq \chi \leq -1$ και $1 \leq \chi \leq 4$, υπολογίζεται η συχνότητα της κάθε χ στον κάθε κύκλο.
6. Για $v_\kappa(\chi)$= το πλήθος των κύκλων στους οποίους η κατάσταση χ συμβαίνει/ εμφανίζεται ακριβώς κ φορές., όπου $\chi = -4, -3, -2, -1, 1, 2, 3, 4$ και $\kappa = 0,1,..,5$, στο $v_5(\chi)$ προσμετρώνται επιπλέον, και οι κύκλοι στους οποίους η χ εμφανίζεται περισσότερες από 5 φορές.
   (Θα πρέπει να ισχύει ότι $\sum_{\kappa=0}^{5} v_\kappa(\chi) = 5$).
7. $\chi^2(obs) = \sum_{\kappa=0}^{5} \frac{(v_\kappa(\chi) - j\pi_\kappa(\chi))^2}{j\pi_\kappa(\chi)}$, όπου οι $\pi_\kappa(\chi)$ επιλέγονται από τον πίνακα:

	$\pi_0(\chi)$	$\pi_1(\chi)$	$\pi_2(\chi)$	$\pi_3(\chi)$	$\pi_4(\chi)$	$\pi_5(\chi)$
χ=1	0.5000	0.2500	0.1250	0.0625	0.0312	0.0312
χ=2	0.7500	0.0625	0.0469	0.0352	0.0264	0.0791
χ=3	0.8333	0.0278	0.0231	0.0193	0.0161	0.0804
χ=4	0.8750	0.0156	0.0137	0.0120	0.0105	0.0733
χ=5	0.9000	0.0100	0.0090	0.0081	0.0073	0.0656
χ=6	0.9167	0.0069	0.0064	0.0058	0.0053	0.0588
χ=7	0.9286	0.0051	0.0047	0.0044	0.0041	0.0531

(οι $\pi_\kappa(\chi)$ είναι η θεωρητική πιθανότητα η κατάσταση χ να συμβεί κ φορές σε έναν κύκλο)

8. Για κάθε κατάσταση χ, $P-value = igamc(\frac{5}{2}, \frac{\chi^2(obs)}{2})$.
   (Επομένως θα υπολογιστούν 8 $P-values$).

Συμπέρασμα: Η ε δεν είναι τυχαία και αποτυγχάνει στο τεστ αν $P-value < 0.01$. Αλλιώς (αν όλες οι $P-values$ είναι ≥ 0.01) η ε είναι τυχαία.

Σημειώσεις:

1. Αν οι τιμές της $\chi^2(obs)$ είναι πολύ μεγάλες, τότε η ακολουθία παρουσιάζει απόκλιση από την αναμενόμενη θεωρητική κατανομή για μια δοσμένη κατάσταση κατά μήκος όλων των κύκλων.
2. Αν για μια υπό εξέταση ακολουθία έχουμε αντιφάσεις στα αποτελέσματα, πχ από τις 8 $P-values$, οι 7 να είναι εντός ορίου και μόλις μια εκτός, καλό θα είναι να εξετάζονται και άλλες ακολουθίες που παράγονται από την ίδια γεννήτρια για να μπορέσουμε να αποφανθούμε για αυτήν (την γεννήτρια).

Παράδειγμα 2.14.1

Για ε= 0110110101 , n=10

(βήμα 1^o)

X= -1, 1, 1, -1, 1, 1, -1, 1, -1, 1.

(βήμα 2^o)

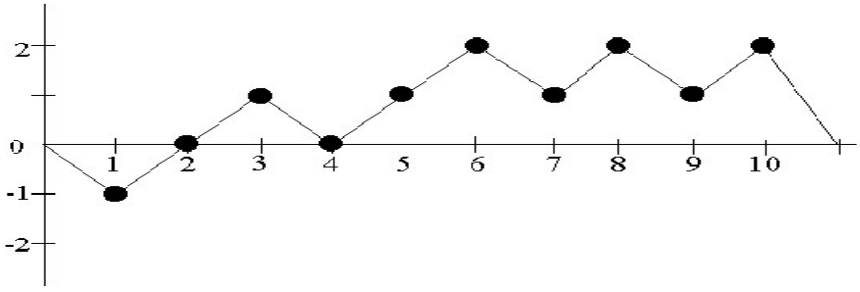
S_1= -1, S_2=0, S_3=1, S_4=0, S_5=1, S_6 = 2, S_7=1, S_8=2, S_9=1, S_{10}=2

S = {-1, 0, 1, 0, 1, 2, 1, 2, 1 ,2}

(βήμα 3^o)

S' = {0, -1, 0, 1, 0, 1, 2, 1, 2, 1 ,2, 0}

Σημείωση: Αν θα θέλαμε να απεικονίσουμε γραφικά τον τυχαίο περίπατο S', θα είχαμε:



(βήμα 4^o)

J=3 (έχουμε 0 στις θέσεις 3, 5 και 12 του S')

Έχουμε 3 κύκλους {0, -1, 0}, {0, 1, 0} και {0, 1, 2, 1, 2, 1, 2, 0}

(βήμα 5)

Κατάσταση χ	$1^{ος}$ κύκλος	$2^{ος}$ κύκλος	$3^{ος}$ κύκλος
-4	0	0	0
-3	0	0	0
-2	0	0	0
-1	1	0	0
1	0	1	3
2	0	0	3
3	0	0	0
4	0	0	0

(βήμα 6^o)

$v_0(-1) = 2$ (η κατάσταση -1 συμβαίνει ακριβώς 0 φορές σε 2 κύκλους),

$v_1(-1) = 1$ (η κατάσταση -1 συμβαίνει ακριβώς 1 φορά σε 1 κύκλου),

$v_2(-1) = v_3(-1) = v_4(-1) = v_5(-1) = 0$ (η κατάσταση -1 συμβαίνει ακριβώς 2, 3, 4, ≥ 5 φορές σε 0 κύκλους),

$v_0(1) = 1$ (η κατάσταση 1 συμβαίνει ακριβώς 0 φορές σε 1 κύκλο),

$v_1(1) = 1$ (η κατάσταση 1 συμβαίνει ακριβώς 1 φορά σε 1 κύκλο),

$v_3(1) = 1$(η κατάσταση 1 συμβαίνει ακριβώς 3 φορά σε 1 κύκλο),

...

για διευκόλυνση του αναγνώστη παραθέτεται ο επόμενος πίνακας μέσω του οποίου μπορεί να βρει όλες τις τιμές για τα $v_k(\chi)$:

Κατ/ση x	Αριθμός Κύκλων στους οποίους εμφανίζεται ακριβώς					
	0 φορές	1 φορά	2 φορές	3 φορές	4 φορές	5 φορές
-4	3	0	0	0	0	0
-3	3	0	0	0	0	0
-2	3	0	0	0	0	0
-1	2	1	0	0	0	0
1	1	1	0	1	0	0
2	2	0	0	1	0	0
3	3	0	0	0	0	0
4	3	0	0	0	0	0

(Βήμα 7^{o})

Όταν χ=1

$\chi^2 = \frac{(1-3(0.5))^2}{3(0.5)} + \frac{(1-3(0.25))^2}{3(0.25)} + \frac{(0-3(0.125))^2}{3(0.125)} + \frac{(1-3(0.0625))^2}{3(0.0625)} + \frac{(0-3(0.0312)^2}{3(0.0312)} + \frac{(0-3(0.0312)^2}{3(0.0312)} =$ 4.333033, όμοια γίνονται οι υπολογισμοί και για τις υπόλοιπες καταστάσεις του χ.

(βήμα 8^{o})

Όταν χ=1, $P - value = igamc\left(\frac{5}{2}, \frac{4.333033}{2}\right) = 0.502529$, όμοια γίνονται οι υπολογισμοί και για τις υπόλοιπες καταστάσεις του χ.

2.15 Random Excursions Variant Test

Ο έλεγχος αυτός εστιάζει στο πόσες φορές εμφανίζεται/συμβαίνει μια συγκεκριμένη κατάσταση σε έναν ¨αθροιστικό¨ τυχαίο περίπατο (βλέπε 2.14). Σκοπός του είναι να ανιχνεύσει αποκλίσεις από τον αναμενόμενο αριθμό επισκέψεων διαφόρων καταστάσεων στον τυχαίο περίπατο. Είναι ουσιαστικά μια σειρά από 18 τεστ (και αντίστοιχα συμπεράσματα), ένα (τεστ και συμπέρασμα) για κάθε μια από τις καταστάσεις -9, -8, ... , -1 και 1, 2, ..., 9. Το τεστ χρησιμοποιεί ως κατανομή αναφοράς την ημι-κανονική κατανομή (η μία της πλευρά είναι διάφορη της κανονικής κατανομής).

Δεδομένα:

1. n=το μήκος της υπό εξέταση ακολουθίας.
2. $\varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n) =$ η υπό εξέταση ακολουθία.
   (Συνιστώμενα μεγέθη: $n \geq 10^6$)

Διαδικασία:

1. χ_i=$2\varepsilon_i$-1, $i = 1, \dots, n$

(Μετατρέπονται τα στοιχεία της ακολουθίας ε που είναι ίσα με 0 σε 1, ενώ αυτά που είναι ίσα με 1 παραμένουν ως έχουν και παράγεται έτσι μια νέα ακολουθία $(\chi_n)_{n=1}^{\infty}$=X).

2. $S_\kappa = \chi_1 + \chi_2 + \cdots + \chi_\kappa, \kappa = 1,2,\dots,n$
   (υπολογίζονται τα μερικά αθροίσματα των διαδοχικά αυξανόμενων υπακολουθιών της X, δηλαδή $S_1 = \chi_1$, ..., $S_n = \chi_1 + \chi_2 + \cdots \chi_n$).
3. $S = \{S_1, \dots, S_n\}$, (η ακολουθία που έχει ως όρους τα μερικά αθροίσματα). $S' = \{0, S_1, \dots, S_n, 0\}$, (προσθέτουμε στην S από ένα 0 στην αρχή και στο τέλος).
4. Για κάθε κατάσταση $\chi \in \mathbb{Z}$ με $-9 \le \chi \le -1$ και $1 \le \chi \le 9$, υπολογίζεται το ξ(χ)= ο συνολικός αριθμός εμφανίσεων της κατάστασης χ κατά μήκος όλων των J κύκλων.
5. Για κάθε κατάσταση χ υπολογίζεται η $P - value = erfc(\frac{|\xi(\chi)-J|}{\sqrt{2J(4|\chi|-2}})$

Συμπέρασμα: Η ε δεν είναι τυχαία και αποτυγχάνει στο τεστ αν $P - value < 0.01$. Αλλιώς (αν όλες οι $P - values$ είναι ≥ 0.01) η ε είναι τυχαία.

Σημείωση: Θα έχουμε συνολικά 18 καταστάσεις και θα αξιολογήσουμε τα αποτελέσματα τους όπως και στο 2.14.

Παράδειγμα 2.15.1

Θα χρησιμοποιήσουμε την ε= 0110110101 του παραδείγματος 2.14.1, n=10

(βήμα 1^o)

X= -1, 1, 1, -1, 1, 1, -1, 1, -1, 1.

(βήμα 2^o)

S_1= -1, S_2=0, S_3=1, S_4=0, S_5=1, S_6 = 2, S_7=1, S_8=2, S_9=1, S_{10}=2

S = {-1, 0, 1, 0, 1, 2, 1, 2, 1 ,2}

(βήμα 3^o)

S' = {0, -1, 0, 1, 0, 1, 2, 1, 2, 1 ,2, 0}

J=3 (έχουμε 0 στις θέσεις 3, 5 και 12 του S')

Έχουμε 3 κύκλους {0, -1, 0}, {0, 1, 0} και {0, 1, 2, 1, 2, 1, 2, 0}

(βήμα 4^o)

$\xi(-1) = 1, \xi(1) = 4, \xi(2) = 3,$

$\xi(-9) = \xi(-8) = \xi(-7) = \xi(-6) = \xi(-5) = \xi(-4) = \xi(-3) = \xi(-2) = \xi(3) = \xi(4) = = \xi(5) = \xi(6) = \xi(7) = \xi(8) = \xi(9) = 0$

(βήμα 5^o)

Για την κατάσταση $\chi = 1$, $P - value = erfc\left(\frac{|4-3|}{\sqrt{6(4|1|-2}}\right) = 0.683091$,

όμοια υπολογίζονται οι $P - values$ και για τις υπόλοιπες 17 καταστάσεις χ.

3 Κατηγοριοποίηση και εισαγωγή στις ψευδοτυχαίες ακολουθίες

Εισαγωγή

Στο κεφάλαιο αυτό θα δοθούν οι βασικοί ορισμοί που αφορούν στις ΓΨΑ και το περιβάλλον τους. Στη συνέχεια θα δοθούν βασικές περιοχές χρήσης τους, με εμβάθυνση στα υπολογιστικά συστήματα καθώς και η κατηγοριοποίηση τους. Θα γίνει εκτενής ανάλυση των γεννητριών ψευδοτυχαίων ακολουθιών σε καταχωρητές ολίσθησης με γραμμική ανάδραση οι οποίοι μας δίνουν τις βασικές ακολουθίες ψευδοτυχαιων ακολουθιών. Έπειτα θα δούμε πως μπορούμε να τις βελτιώσουμε σημαντικά χρησιμοποιώντας μη γραμμικές πράξεις. Για να καταλήξουμε στην ανάλυση γνωστών ΓΨΑ, η κάθε μια από τις οι οποίες αποτελεί βασικό αντιπρόσωπο στον τομέα της.

3.1 Βασικοί Ορισμοί

Ορισμός 3.1.1: Ως ακολουθία τυχαίων αριθμών ορίζουμε μια ακολουθία της οποίας οι όροι ικανοποιούν τα κάτωθι:

a. οι τιμές τους (των όρων της) είναι ομοιόμορφα κατανεμημένες σε ένα καθορισμένο διάστημα ή σύνολο,
b. βασιζόμενοι στην (τρέχουσα) τιμή ενός όρου της ή σε τιμές προηγούμενων όρων της, γενικότερα, είναι αδύνατο να προβλέψουμε την τιμή κάποιου επόμενου όρου.

Ορισμός 3.1.2: Μια ακολουθία αριθμών καλείται ψευδοτυχαία όταν:

a. περνά όλους τους γνωστούς στατιστικούς ελέγχους περί τυχαιότητας (στατιστική απαίτηση).
b. και είναι απρόβλεπτη. Δηλαδή, είναι υπολογιστικά αδύνατο χρησιμοποιώντας ένα τμήμα της να καθοριστεί το αμέσως επόμενο στοιχείο της (κρυπτογραφική απαίτηση).

Ορισμός 3.1.3: Μια ακολουθία αριθμών καλείται πραγματικά τυχαία όταν:

a. ικανοποιεί όλες τις απαιτήσεις του ορισμού 3.2 (πληροί, δηλαδή, όλες τις προϋποθέσεις μιας ψευδοτυχαίας ακολουθίας αριθμών) και επιπλέον,
b. δεν μπορεί να αναπαραχθεί με αξιοπιστία.

Ορισμός 3.1.4: Με τον όρο γεννήτρια τυχαίας ακολουθίας δυαδικών ψηφίων (RBG/ Random Bit Generator) αναφερόμαστε σε έναν μηχανισμό ή αλγόριθμο, η έξοδος του

οποίου είναι μια ακολουθία δυαδικών ψηφιών τα οποία είναι στατιστικά ανεξάρτητα και αμερόληπτα.

Βασική σημείωση: Όπως έχει ήδη αναφερθεί στην παρούσα επικεντρωνόμαστε στις δυαδικές ακολουθίες πεπερασμένου μήκους που έχουν εφαρμογή στα σύγχρονα κρυπτογραφικά συστήματα. Οπότε αναφερόμαστε -ουσιαστικά- σε (πεπερασμένες) ακολουθίες πραγματικά τυχαίων και ψευδοτυχαίων δυαδικών ψηφίων (TRBSs/True Random Binary Sequences και PRBSs/Peudo Random Binary Sequences) και στις αντίστοιχες γεννήτριες τους. Στις οποίες, όμως, μπορούμε να ανάγουμε και τις ακολουθίες τυχαίων αριθμών, υπό την έννοια ότι:

a. χρησιμοποιώντας μια RBG παράγεται μια RBS από την οποία μπορούμε να εξάγουμε έναν τυχαίο αριθμό και επαναλαμβάνοντας τη διαδικασία με διάφορες RBSs να δημιουργήσουμε μια τυχαία ακολουθία αριθμών (ομοιόμορφα κατανεμημένων σε ένα διάστημα).
b. από κάθε (πεπερασμένη) ακολουθία τυχαίων αριθμών μπορούμε να μεταβούμε σε μια (πεπερασμένη) RBS πχ αν στη θέση κάθε αριθμού: παίρνουμε απλά τη δυαδική του αναπαράσταση, ή παίρνουμε το μικρότερο σε αξία bit[4] της δυαδικής του αναπαράστασης (διαδικασία η οποία προτιμάται τις περισσότερες φορές), ή παίρνουμε τον $log_2 n$ (όπου n το μήκος της δυαδικής του αναπαράστασης - τεχνική που χρησιμοποιείται συχνά σε σύγχρονες εφαρμογές) κ.α.

   Παραδείγματος χάριν,

   a. για να παράγουμε έναν τυχαίο ακέραιο στο διάστημα $[0, n]$, μπορούμε να χρησιμοποιήσουμε πρώτα μια RBG για να παράγουμε μια τυχαία δυαδική ακολουθία μήκους $floor(n) + 1$ και να την μετατρέψουμε σε ακέραιο αριθμό (αν είναι $> n$ τον απορρίπτουμε και επαναλαμβάνουμε τη διαδικασία, αλλιώς τον δεχόμαστε),
   b. δείτε την διαδικασία της ΓΨΑ BBS που αναλύεται παρακάτω (3.10).

Επομένως, στην συνέχεια με τον όρο ακολουθία θα αναφερόμαστε σε μια πεπερασμένου μήκους δυαδική ακολουθία (η οποία υπολογιστικά αποτελεί τον πυρήνα όλων των άλλων).

Ορισμός 3.1.5: Με τον όρο γεννήτρια τυχαίας ακολουθίας (ΓΤΑ/RNG-Random Number Generator) αναφερόμαστε σε έναν μηχανισμό ή αλγόριθμο, η έξοδος του οποίου είναι μια ακολουθία αριθμών οι οποίοι είναι στατιστικά ανεξάρτητοι και μη προβλέψιμοι.

[4] Θα εξηγηθεί στην 3.6

Οι Γ.Τ.Α ταξινομούνται σε δυο βασικές κατηγορίες:

a. τις Γεννήτριες Ψευδοτυχαίων Ακολουθιών (ΓΨΑ/PRN.Gs-Pseudo Random Number Generators) και
b. τις Γεννήτριες Πραγματικά Τυχαίων Ακολουθιών (ΓΠΤΑ/TRNGs-True Random Number Generators).

Ορισμός 3.1.6: Με τον όρο ΓΨΑ καλείται ένας ντετερμινιστικός αλγόριθμος, που δέχεται μια πραγματικά τυχαία ακολουθία μήκους m ως είσοδο, συνήθως αναφέρεται ως σπόρος (seed), και δίνει ως έξοδο μια ακολουθία μήκους $l >> m$ η οποία ¨δείχνει να είναι¨ τυχαία, αλλά στην πραγματικότητα δεν είναι και με τη σειρά της καλείται ψευδοτυχαία ακολουθία.

Ουσιαστικά μια ΓΨΑ επεκτείνει μια πραγματικά τυχαία ακολουθία σε μια με πολύ μεγαλύτερο μήκος. Με σκοπό να μην μπορεί κάποιος (ο αντίπαλος με όρους κρυπτογραφίας) να την ξεχωρίσει από πραγματικά τυχαίες ακολουθίες ιδίου μήκους.
Επιπλέον, αν σε μια ΓΨΑ δοθεί ως είσοδος ο ίδιος σπόρος, τότε θα δώσει ως έξοδο την ίδια ακριβώς ακολουθία κάθε φορά.

Ένα άλλο χαρακτηριστικό μιας ΓΨΑ είναι ότι η ακολουθία που παράγει από ένα σημείο και μετά αρχίζει να επαναλαμβάνεται. Παράγει, δηλαδή, μια περιοδική ακολουθία (δες ορισμό 1.8.1), και ένα κριτήριο για το πόσο 'καλή' είναι η γεννήτρια είναι το μέγεθος της περιόδου της.

Ορισμός 3.1.7: Με τον όρο ΓΠΤΑ αναφερόμαστε σε έναν μηχανισμό ή αλγόριθμο, η έξοδος του οποίου είναι πλήρως βασισμένη σε μη αιτιοκρατικά φαινόμενα, τα οποία είναι και η πηγή τυχαιότητας του.

Για τέτοιου είδους φαινόμενα χρησιμοποιούμε συνήθως μια τυχαία φυσική διαδικασία (πχ την βροχή) ή ακόμα και τυχαία ερεθίσματα της καθημερινής μας ζωής (πχ την ουρά σε μια τράπεζα). Αυτή η φυσική διαδικασία καθορίζει πλήρως την έξοδο μιας ΓΠΤΑ (και άρα δεν έχει καμία εξάρτηση από κάποια αρχική τιμή -σε αντίθεση με μια ΓΨΑ). Όπως είναι εύκολα αντιληπτό το να δημιουργηθεί ένας μηχανισμός ή ένας αλγόριθμος που να χρησιμοποιεί αποτελεσματικά μια τέτοια πηγή τυχαιότητας φαντάζει αρκετά δύσκολο. Αλλά ακόμα και μετά την ανάπτυξη του μπορεί να προκληθεί δυσλειτουργία στην γεννήτρια από εξωτερικά φαινόμενα. Άρα δεν αρκεί μόνο κάποιος αρχικός έλεγχος της, αλλά επιβάλλονται τακτικοί στατιστικοί έλεγχοι, για να είμαστε σίγουροι για την εύρυθμη λειτουργία της (σημαντικό μειονέκτημα σε σχέση με τις ΓΨΑ).

Στις κρυπτογραφικές εφαρμογές (όπως και σε πολλές άλλες) οι ΓΨΑ προτιμούνται έναντι των ΓΠΤΑ. Λόγω του ότι μια ΓΨΑ:

- Αν δεχθεί την ίδια είσοδο παράγει ακριβώς την ίδια ακολουθία κάθε φορά (εξαιρετικά σημαντικό στην κρυπτογραφία).
- Μπορεί εύκολο να ενσωματωθεί/εγκατασταθεί και να εκτελεστεί.
- Εκτελείται γρήγορα.
- Απαιτεί μικρή υπολογιστική ισχύ.

Οι ΓΨΑ δεν έχουν μόνο πλεονεκτήματα αλλά μπορούν να παρουσιάσουν μειονεκτήματα τα οποία με τη σειρά τους θα προκαλέσουν αποτυχία της γεννήτριας σε κάποιο/α από τα test του Κεφαλαίου 2 (ή σε άλλους ελέγχους αξιολόγησης στους οποίους μπορεί να υποβληθούν). Τέτοια μειονεκτήματα μπορεί να είναι:

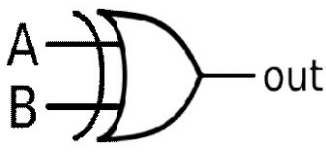
- Οι παραγόμενοι αριθμοί να μην είναι ομοιόμορφα κατανεμημένοι.
- Διαφορετικές αποστάσεις ανάμεσα σε τιμές που εμφανίζονται σε σχέση με αυτές που θα περιμέναμε από μια (πραγματικά) τυχαία ακολουθία.
- Διαδοχικοί όροι της ακολουθίας να παρουσιάζουν συσχέτιση.
- Οι παραγόμενοι αριθμοί της ακολουθίας να έχουν 'κακή' διαστατική κατανομή.
- Η ύπαρξη 'αδύναμων' σπόρων, αρχικών τιμών δηλαδή από τους οποίους παράγονται ακολουθίες με μικρότερη από την αναμενόμενη περίοδο.

3.2 Πρόσθεση στον $\mathbb{Z}_2$ και η λογική πύλη XOR

Η πύλη XOR εκτελεί την πράξη XOR (αποκλειστικού ή) μεταξύ των εισόδων της. Στην άλγεβρα Boole παριστάνεται με το σύμβολο " $\oplus$ " .Έτσι αν η πύλη δέχεται ως δυο εισόδους A, B δίνει έξοδο Γ, σύμφωνα με τη σχέση

$\Gamma = AB' + A'B = A \oplus B.$

Και το κυκλωματικό σχήμα για την XOR δυο εισόδων είναι:



Ενώ ο πίνακας αλήθειας της είναι:

Είσοδοι		Έξοδος
A	B	A XOR B
0	0	0
0	1	1
1	0	1
1	1	0

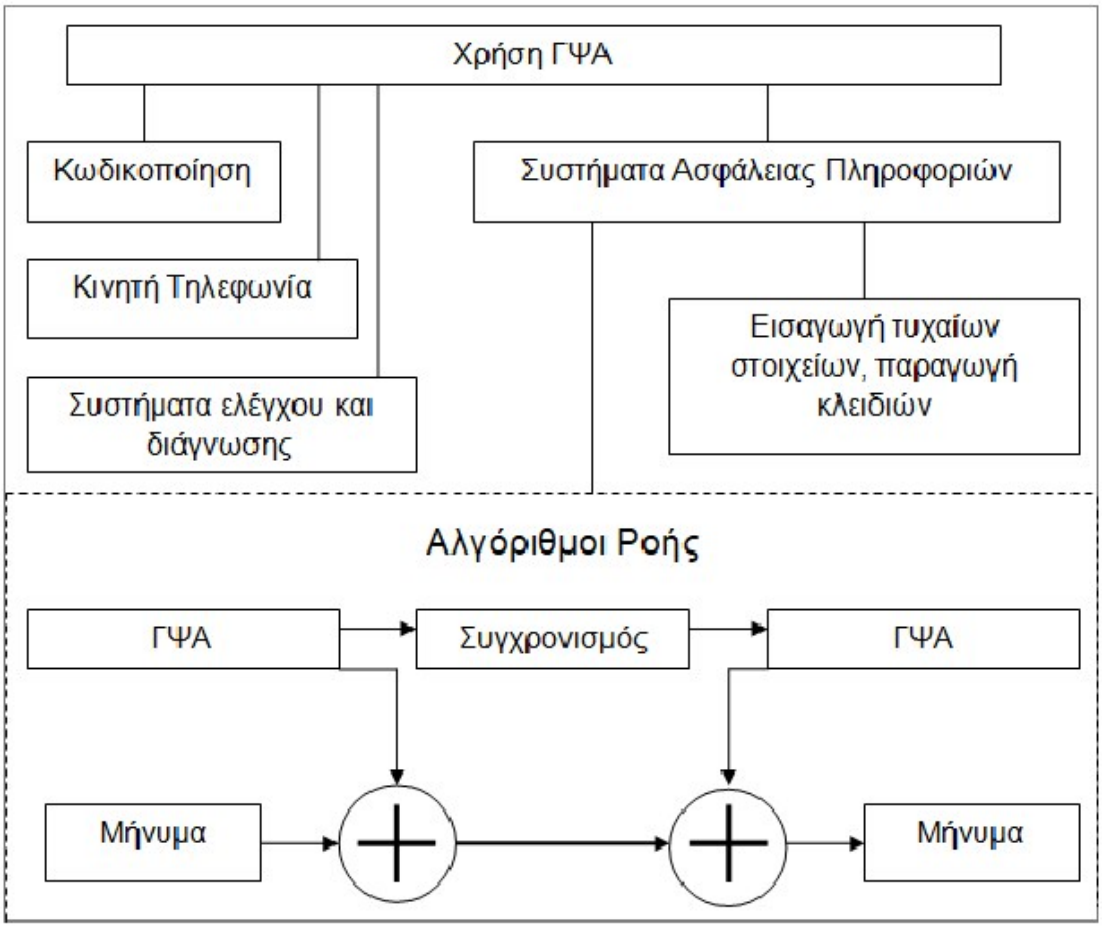
Όπως είναι εύκολα αντιληπτό η λογική πύλη XOR μας δίνει τα ίδια αποτελέσματα με την πρόσθεση $\boldsymbol{modulo}\ 2$ στο $\mathbb{Z}_2$ (για τις ίδιες εισόδους). Για τον λόγο αυτό στην πρόσθεση στοιχείων δυαδικών ακολουθιών πολλές φορές χρησιμοποιείται κατευθείαν η λογική πύλη XOR, αντί της πρόσθεσης (κυρίως στους καταχωρητές ολίσθησης), σύμφωνα με τον κοινό πίνακα αλήθειας:

$\oplus$	0	1
0	0	1
1	1	0

3.3 Ανάλυση της χρήσης των ΓΨΑ σε συστήματα υπολογιστών

Μερικές από συνηθέστερες χρήσεις ΓΨΑ σε συστήματα υπολογιστών είναι οι κάτωθι:

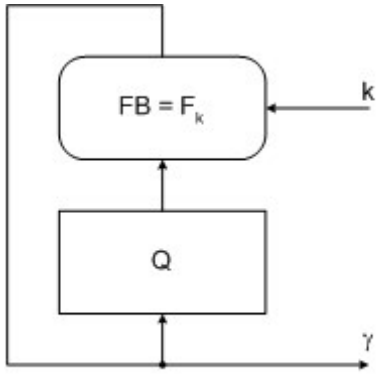
- Παραγωγή ακολουθίας με σκοπό την κρυπτογράφηση ροής.
- Παραγωγή μυστικών κλειδιών σε συστήματα ασφαλείας πληροφοριών.
- Προσομοίωση μιας τυχαίας πηγής δράσης για συστήματα μοντελοποίησης.
- Κατακερματισμός δεδομένων.
- Παραγωγή CRC κωδικών.
- Έλεγχος διαφόρων μηχανισμών και λογισμικών, καθώς και ο αυτοέλεγχος τους..
- Εισαγωγή της αβεβαιότητας στη λειτουργία των συστημάτων ασφαλείας.
- Κωδικοποίηση δεδομένων όταν μεταδίδονται σε διαφορετικά κανάλια επικοινωνίας.



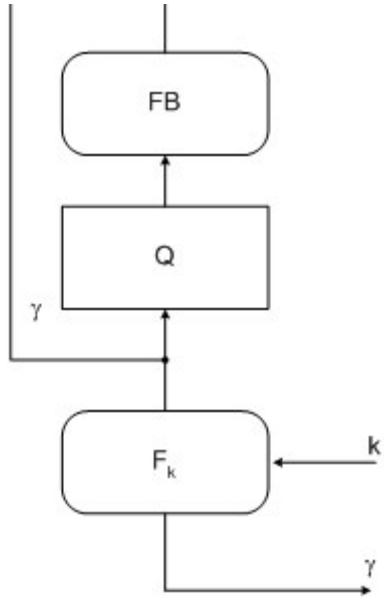
3.4 Κατηγοριοποίηση ΓΨΑ

Στην παρούσα ενότητα θα γίνει αναφορά στις δυο βασικές κατηγορίες στις οποίες χωρίζονται οι ΓΨΑ, καθώς και σε απατήσεις που υπάρχουν προς αυτές. Πρώτα, όμως, θα αναφερθούν δυο γενικοί τρόποι κατασκευής μιας ΓΨΑ, παρέχοντας έτσι στον αναγνώστη μια αναγνωριστική επαφή που θα συμβάλει στην κατανόηση των επομένων. Αναλυτική παρουσίαση γεννητριών υπάρχει στις επόμενες ενότητες.

Στα σχήματα α) και β) παραθέτονται δυο τρόποι σύνθεσης μια ΓΨΑ, οι οποίοι χρησιμοποιούν μια μη γραμμική συνάρτηση F_K. Ο πρώτος με μη γραμμική ανάδραση εσωτερικής λογικής (λειτουργία *OFB – Output FeedBack*) και ο δεύτερος με μη γραμμική εξωτερική λογική (λειτουργία μετρητή *Counter*). Στον πρώτο η F_K.χρησιμοποιείται μέσα στο κύκλωμα ανάδρασης, ενώ στον δεύτερο είναι σχεδιασμένη σε διπλό επίπεδο -στο οποίο η άσκηση του πρώτου επιπέδου (ως προς τον μετρητή) περιλαμβάνει κατά κανόνα την εξασφάλιση της μέγιστης περιόδου για δεδομένο μήκος N του καταχωρητή Q.



Σχήμα α



Σχήμα β

όπου, Q – Στοιχείο μνήμης γεννήτριας,

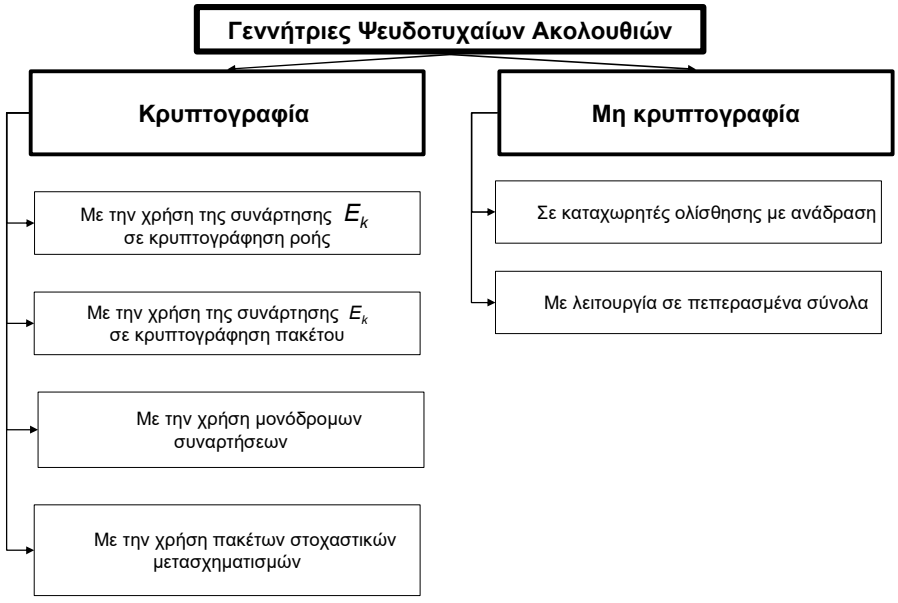
FB – γραμμική ή μη γραμμική συνάρτηση ανάδρασης,

F_k – μη γραμμική συνάρτηση,

k – κλειδί,

γ_i – Στοιχείο εξερχόμενων ακολουθιών

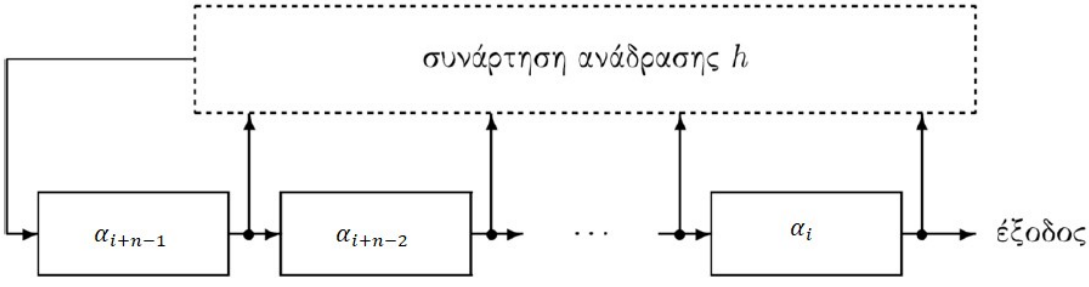
Οι ΓΨΑ χωρίζονται σε δυο βασικές κατηγορίες: στις κρυπτογραφικές και στις μη κρυπτογραφικές. Η διάκριση ανάμεσα στις δυο κατηγορίες έγκειται στην απαίτηση που έχουμε ως προς τις κρυπτογραφικές συναρτήσεις, που χρησιμοποιούν οι κρυπτογραφικές γεννήτριες, να μην μπορούν να αντιστραφούν (εντός πλαισίων που θα έκαναν ευάλωτη την πληροφορία μας), έτσι ώστε να μην μπορέσει ο αντίπαλος να προβεί σε 'σπάσιμο' της γεννήτριας. Υπάρχει, δηλαδή, η επιπλέον απαίτηση της πολυπλοκότητας ως προς το σπάσιμο της γεννήτριας.



Για να μπορέσει όμως να χρησιμοποιηθεί μια ΓΨΑ από ένα σύστημα ασφαλείας πληροφοριών υπάρχουν οι εξής απαιτήσεις:

- Επαρκές μήκος περιόδου επανάληψης των παραγόμενων ακολουθιών..
- Ικανοποίηση στατιστικών κριτηρίων των παραγόμενων ακολουθιών.
- Δυνατότητα λογισμικής υλοποίησης και παραγωγής ακολουθιών με γρήγορους ρυθμούς.
- Μη προβλεψιμότητα.

3.5 ΓΨΑ με βάση καταχωρητές ολίσθησης με ανάδραση (Feedback Shift Register Generaotors FSRs)



Οι καταχωρητές ολίσθησης/συστήματα καταγραφής μετατόπισης ανήκουν στην κατηγορία των πεπερασμένων μηχανών. Όπως είπαμε θα τους μελετήσουμε στο $\mathbb{F}_2$, η γενική αναφορά στο $\mathbb{F}_q$ έχει γίνει στο 1.8. Ένας καταχηρητής με μήκος n στο $\mathbb{F}_2$ αποτελείται από

n θέσεις μνήμης/βαθμίδες. Κάθε βαθμίδα περιέχει το στοιχείο 0 ή το στοιχείο 1. Κάθε παλμός του ρολογιού (κάθε $i \geq 0$) μετατοπίζει το περιεχόμενο της κάθε βαθμίδας σε αυτήν που βρίσκεται δεξιά της. Το κενό περιεχομένου που δημιουργείται στην βαθμίδα που βρίσκεται τέρμα αριστερά καλύπτεται από το αποτέλεσμα της συνάρτησης ανάδρασης h (η οποία χρησιμοποιεί ως ορίσματα τα περιεχόμενα των βαθμίδων πριν τον τρέχοντα παλμό του ρολογιού). Επομένως κάθε ψευδοτυχαία ακολουθία που παράγεται από έναν τέτοιο καταχωρητή ικανοποιεί την αναδρομική σχέση:

$$\alpha_{i+n} = h(\alpha_i, \alpha_{i+1}, \dots, \alpha_{i+n-1}), i \geq 0$$

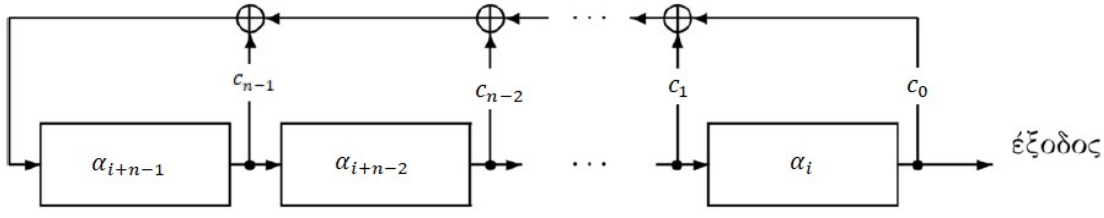
Κάθε θέση μνήμης του καταχωρητή αντιστοιχεί σε ένα flip-flop, η συνάρτηση ανάδρασης h είναι μια λογική συνάρτηση (Boolean function) με n μεταβλητές και η παραγόμενη ακολουθία $(\alpha)_i$ αποτελεί μια ψευδοτυχαία ακολουθία (αν η h είναι μη μηδενικού σταθερού όρου). Ανάλογα με το είδος της συνάρτησης ανάδρασης h οι FSRs χωρίζονται σε γραμμικούς και μη-γραμμικούς.

3.6 ΓΨΑ με βάση γραμμικούς καταχωρητές ολίσθησης με ανάδραση (Linear Feedback Shift Register Generaotors LFSRs)

Οι καταχωρητές ολίσθησης με ανάδραση των οποίων η συνάρτηση ανάδρασης h είναι γραμμική ονομάζονται γραμμικοί καταχωρητές ολίσθησης με ανάδραση (LFSR). Στους LFSR, δηλαδή, η συνάρτηση ανάδρασης h έχει την μορφή:

$$\alpha_{i+n} = h(\alpha_i, \alpha_{i+1}, \dots, \alpha_{i+n-1}) = c_{n-1}\alpha_{i+n-1} + c_{n-2}\alpha_{i+n-2} + \cdots + c_0\alpha_i, \quad (1)$$

όπου $c_j, \alpha_j \in \{0,1\}$



και το χαρακτηριστικό πολυώνυμο ενός LFSR είναι το:

$$f(\chi) = \chi^n - c_{n-1}\chi^{n-1} - \cdots - c_0 \in \mathbb{F}_2$$

και λόγω της αριθμητικής στον $\mathbb{F}_2$:

$$f(\chi) = \chi^n + c_{n-1}\chi^{n-1} + \cdots + c_0 \in \mathbb{F}_2$$

ενώ το πολυώνυμο ανάδρασης, το οποίο επίσης προσδιορίζει μονοσήμαντα έναν LFSR και άρα χρησιμοποιείται και αυτό (κάποιες φορές) για να τον περιγράψει, είναι το:

$$f'(\chi) = 1 - c_{n-1}\chi - \cdots - c_0\chi^n = 1 + c_{n-1}\chi + \cdots + c_0\chi^n \in \mathbb{F}_2$$

Και αντιστρόφως αν σε μια ακολουθία όλοι οι όροι της ικανοποιούν την *(1)* τότε το $f(\chi)$,

που αναφέρθηκε προηγουμένως, καλείται χαρακτηριστικό πολυώνυμο της ακολουθίας. Και όπως μια ακολουθία έχει, γενικά, πολλά χαρακτηριστικά πολυώνυμα, έτσι και μια ακολουθία μπορεί να παραχθεί από διάφορους LFSRs.

Η παραγόμενη ακολουθία είναι περιοδική αν και μόνο αν $f(0) \neq 0$, αν και μόνο αν, δηλαδή, ο σταθερός όρους του χαρακτηριστικού της πολυωνύμου είναι διάφορος του μηδενός (και άρα λόγο $\mathbb{F}_2$ ίσος με 1). Θα ασχοληθούμε μόνο με τέτοιου είδους περιπτώσεις γιατί μια ψευδοτυχαία ακολουθία πρέπει να είναι περιοδική. Άρα θα παράγεται από LFSR με χαρακτηριστικό πολυώνυμο: $f(\chi) = \chi^n + c_{n-1}\chi^{n-1} + \cdots + 1 \in \mathbb{F}_2$

Η βασική ακολουθία ΓΨΑ είναι ακολουθία που παράγεται από έναν LFSR. Όπως σε κάθε ΓΨΑ έτσι και εδώ είναι απαραίτητη μια αρχική κατάσταση (ο σπόρος). Ο σπόρος όπως είναι προφανές θα είναι μια (δυαδική) ακολουθία/ αρχική κατάσταση μήκους n, αλλά όχι η $\{0,0,\dots,0\}$, γιατί σε αυτή την περίπτωση ένας LFSR θα παραμείνει συνεχώς σε αυτήν την κατάσταση και θα παράγει μόνο 0.

Αν όμως σε οποιονδήποτε LFSR (με χαρακτηριστικό πολυώνυμο f αντίστοιχα) δοθεί ως σπόρος η $\{1,0,0,\dots,0\}$, τότε η ακολουθία που θα παραχθεί θα έχει τη μέγιστη δυνατή περίοδο ακολουθίας που παράγεται από την συγκεκριμένο LFSR και καλείται ακολουθία κρουστικής απόκρουσης. Κάθε άλλη ακολουθία που παράγεται από τον συγκεκριμένο καταχωρητή θα έχει περίοδο η οποία: είτε θα είναι ίση ,είτε θα διαιρεί (ακριβώς), την περίοδο της ακολουθίας κρουστικής απόκρουσης. Από την άλλη, η περίοδος της ακολουθίας κρουστικής απόκρουσης είναι ίση με την τάξη του χαρακτηριστικού πολυωνύμου f ($ord(f) = min\{\kappa \in Z: f \setminus (\chi^\kappa - 1)\}$, βλέπε κεφάλαιο 1). Αν το f είναι μη ανάγωγο στο $\mathbb{F}_2$ και $deg(f) = n$, τότε $ord(f) \setminus (2^n - 1)$. Ισχύει ότι $ord(f) = 2^n - 1 \Leftrightarrow f$ πρωταρχικό πολυώνυμο στο $\mathbb{F}_2$.

Έστω ακολουθία (α_n) που όλοι οι όροι της ικανοποιούν την *(1)*, τότε υπάρχει μονικό πολυώνυμο $m(\chi) \in \mathbb{F}_2[\chi]$ τέτοιο ώστε:

ένα μονικό πολυώνυμο $f(\chi) \in \mathbb{F}_2[\chi]$ είναι χαρακτηριστικό πολυώνυμο της $(\alpha_n) \Leftrightarrow$
$m(\chi) \setminus f(\chi)$.

Τότε το $m(\chi)$ καλείται ελάχιστο πολυώνυμο της ακολουθίας (α_n). Επομένως η συνάρτηση ανάδρασης h με το μικρότερο μήκος ενός LFSR που παράγει μια ακολουθία (α_n), περιγράφεται μέσω του ελαχίστου πολυωνύμου $m(\chi)$ της (α_n). Αν η (α_n) έχει ελάχιστο πολυώνυμο $m(\chi)$ τότε η περίοδος της ισούται με $ord(m(x))$. Επιπροσθέτως, σε περίπτωση που το χαρακτηριστικό πολυώνυμο μιας ακολουθίας είναι πρωταρχικό - και άρα θα η ακολουθία θα έχει περίοδο $2^n - 1$ - τότε το ελάχιστο πολυώνυμο της ταυτίζεται με το χαρακτηριστικό πολυώνυμο της.

Οι LFSRs που έχουν ένα πρωταρχικό πολυώνυμο ως χαρακτηριστικό (πολυώνυμο) ονομάζονται πρωταρχικοί LFSRs και οι ακολουθίες που παράγονται από αυτούς ονομάζονται ακολουθίες μεγίστου μήκους.

Επομένως ένας LFSR δίνει πάντα ακολουθίες με μέγιστη περίοδο ίση με $2^n - 1$ αν και μόνο αν το χαρακτηριστικό του πολυώνυμο είναι πρωταρχικό (ανεξάρτητα από την τιμή του σπόρου). Αν, δηλαδή, το χαρακτηριστικό του πολυώνυμο ικανοποιεί τα κάτωθι:

a. είναι ανάγωγο και deg(f)=n
b. $f \setminus (\chi^\lambda - 1) \Leftrightarrow f \setminus (\chi^\lambda + 1)$, για $\lambda = 2^n - 1$
c. $f \nmid (\chi^\lambda - 1) \Leftrightarrow f \nmid (\chi^\lambda + 10$, για $\lambda < 2^n - 1$

Παρακάτω παρουσιάζονται κάποια από τα πρωταρχικά πολυώνυμα έως και $73^{ου}$ βαθμού.

\

Βαθμός (n)	Πολυώνυμο	Βαθμός (n)	Πολυώνυμο
2, 3, 4, 6, 7, 15, 22, 60, 63	$x^n + x + 1$	12	$x^n + x^7 + x^4 + x^3 + 1$
5, 11, 21, 29, 35	$x^n + x^2 + 1$	33	$x^n + x^{13} + 1$
8, 19, 38, 43	$x^n + x^6 + x^5 + x + 1$	34	$x^n + x^{15} + x^{14} + x + 1$
9, 39	$x^n + x^4 + 1$	36	$x^n + x^{11} + 1$
10, 17, 20, 25, 28, 31, 41, 52	$x^n + x^3 + 1$	37	$x^n + x^{12} + x^{10} + x^2 + 1$
13, 24, 45, 64	$x^n + x^4 + x^3 + x + 1$	40	$x^n + x^{21} + x^{19} + x^2 + 1$
14, 16	$x^n + x^5 + x^4 + x^3 + 1$	42	$x^n + x^{23} + x^{22} + x + 1$
18, 57	$x^n + x^7 + 1$	46	$x^n + x^{21} + x^{20} + x + 1$
23, 47	$x^n + x^5 + 1$	54	$x^n + x^{37} + x^{36} + x + 1$
26, 27	$x^n + x^{12} + x^{11} + x + 1$	55	$x^n + x^{24} + 1$
30, 51, 53, 61, 70	$x^n + x^{16} + x^{15} + x + 1$	58	$x^n + x^{19} + 1$
32, 48	$x^n + x^{28} + x^{27} + x + 1$	65	$x^n + x^{18} + 1$
44, 50	$x^n + x^{27} + x^{26} + x + 1$	69	$x^n + x^{29} + x^{27} + x^2 + 1$
49, 68	$x^n + x^9 + 1$	71	$x^n + x^6 + 1$
56, 59	$x^n + x^{22} + x^{21} + x + 1$	72	$x^n + x^{53} + x^{47} + x^6 + 1$
66, 67, 74	$x^n + x^{10} + x^9 + x + 1$	73	$x^n + x^{25} + 1$

Ακολουθούν μήκη ακολουθιών που παράγονται από πρωταρχικούς LFSRs 2ου έως και 32ου βαθμού.

Βαθμός πολυωνύμου	Μήκος της ακολουθίας	Βαθμός πολυωνύμου	Μήκος της ακολουθίας
2	3	18	262.143
3	7	19	524.287
4	15	20	1.048.575
5	31	21	2.097.151
6	63	22	4.194.303
7	127	23	8.388.607
8	255	24	16.777.215
9	511	25	33.554.431
10	1.023	26	67.108.863
11	2.047	27	134.217.727
12	4.095	28	268.435.455
13	8.191	29	536.870.911
14	16.383	30	1.073.741.823
15	32.767	31	2.147.483.647
16	65.535	32	4.294.967.295
17	131.071		

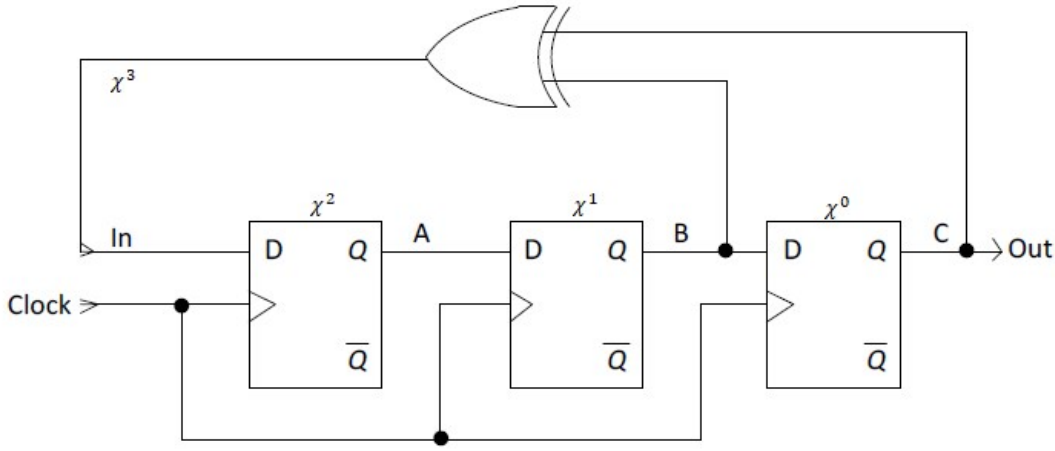
Κύρια οφέλη των LFSR είναι:

- οι ακολουθίες που παράγουν έχουν καλές στατιστικές ιδιότητες.
- ευκολία υλοποίησης σε υλικό και λογισμικό
- μέγιστη αποδοτικότητα για την ταχύτητα εκτέλεσης
- μπορούν να ανταπεξέλθουν σε ειδικές απαιτήσεις της ασφάλειας πληροφοριών (όπως σχηματισμό ακολουθίας με προκαθορισμένη περίοδο ή τυχαίου μήκους, όπως κατασκευή ΓΨΑ που ικανοποιεί τις ιδιότητες του αυτοελέγχου ή με τυχαίο κανόνα κατανομής κ.α.) με κατάλληλη επιλογή LFSR.

Ανάλογα με τον τρόπο σύνδεσης της λογικής πύλης XOR στο κύκλωμα του καταχωρητή ολίσθησης, οι LFSRs χωρίζονται σε δυο κατηγορίες: α) τους εξωτερικούς (externals) ή αλλιώς Fibonnacci ή many-to-one LFSRs και β) τους εσωτερικούς (internals) ή αλλιώς Galois ή one-to-many LFSRs. Στα Fibonnacci οι έξοδοι των flip-flops συνδέονται στις εισόδους της λογικής πύλης και η έξοδος της πύλης συνδέεται με τη σειρά της πίσω στην είσοδο. Ενώ στα Galois η λογική πύλη συνδέεται ανάμεσα στα flip flops.

Για παράδειγμα ένα πρωταρχικό Fibonnaci LFSR με $n = 3$ και χαρακτηριστικό πολυώνυμο $\chi^3 + \chi + 1$ (3-bit Fibonacci Primitive LFSR) ακολουθεί την παρακάτω

συνδεσμολογία:



Στο οποίο αν δοθεί ως αρχική κατάσταση (σπόρος) η 111, θα περάσει από τις εξής καταστάσεις (μέχρι να διπλώσει και να αρχίσει να επαναλαμβάνεται):

011

001

100

010

101

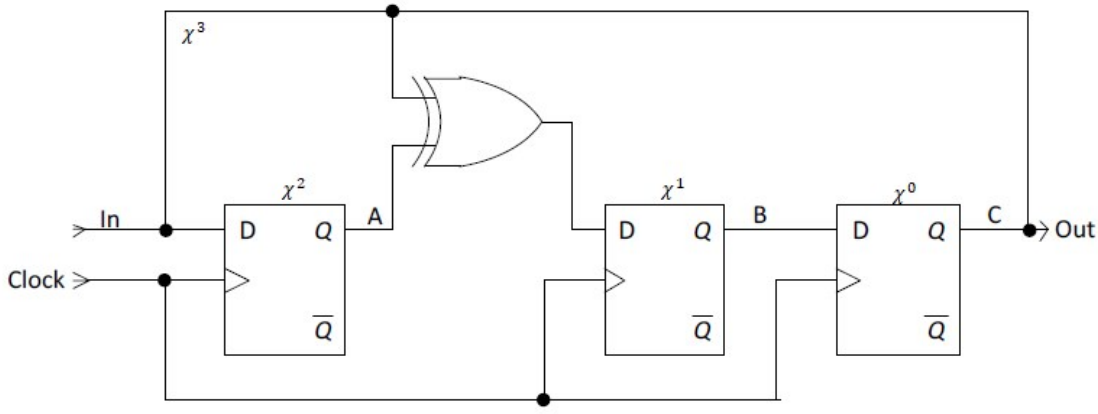
110

111 (ξεκινάει η επανάληψη οπότε σταματάμε)

Ως εξαγόμενη ακολουθία παίρνουμε αυτή του λιγότερου σημαντικού bit (less significant bit) κάθε κατάστασης, που ως λιγότερο σημαντικό bit θεωρείται το δεξιότερο ψηφίο της δυαδικής αναπαράστασης κάθε κατάστασης/ακολουθίας. Επομένως η έξοδος είναι: 1110010, η οποία έχει μήκος $2^3 - 1 = 7$ (αφού προέρχεται από πρωταρχικό LFSR). Πιο συνοπτικά:

A	B	C
1	1	1
0	1	1
0	0	1
1	0	0
0	1	0
1	0	1
1	1	0
1	1	1

Ενώ ένα πρωταρχικό Galois LFSR με $n = 3$ και χαρακτηριστικό πολυώνυμο $\chi^3 + \chi + 1$ (3-bit Galois Primitive LFSR) ακολουθεί την παρακάτω συνδεσμολογία:



Στο οποίο αν δοθεί ως σπόρος ο 111, έχουμε συνοπτικά:

A	B	C
1	1	1
1	0	1
1	0	0
0	1	0
0	0	1
1	1	0
0	1	1
1	1	1

Και άρα έχουμε ως έξοδο: 1100101. Όπως βλέπουμε τα δυο LFSR παρότι έχουν ίδιο χαρακτηριστικό πολυώνυμο και σπόρο δίνουν διαφορετική έξοδο, Αν όμως μετατοπίσουμε το ρολόι κατά έναν παλμό οι δυο έξοδοι θα ταυτιστούν.

Ιδιαιτέρως σημαντικό είναι το γεγονός ότι: για οποιαδήποτε ακολουθία η οποία έχει παραχθεί με οποιονδήποτε τρόπο (πχ από μη γραμμικό FSR), μπορούμε να βρούμε πάντα κάποιους LFSR που επίσης την παράγουν. Το μήκος του μικρότερου LFSR που παράγει μια ακολουθία ονομάζεται γραμμική πολυπλοκότητα της ακολουθίας. Η γραμμική πολυπλοκότητα μιας ακολουθίας είναι πάρα πολύ σημαντική στις κρυπτογραφικές εφαρμογές λόγω των κάτωθι:

- Μέσω του αλγορίθμου Berlekamp-Massey[5] (1969) μπορεί να υπολογιστεί σε πολυωνυμικό χρόνο ελάχιστος LFSR που παράγει την ακολουθία.
- Αν n είναι το μήκος μιας ακολουθίας και για την γραμμική της πολυπλοκότητα L ισχύει: $L < {}^{n}/_{2}$,τότε ο μικρότερος LFSR μήκους L που την παράγει είναι μοναδικός. Και μπορεί να βρεθεί (ο ο μικρότερος LFSR μήκους L που την παράγει) μέσω του αλγορίθμου Berlekamp-Massey, αν (με οποιονδήποτε τρόπο) έχουν γίνει γνωστά $2L$ διαδοχικά bits της ακολουθίας.

[5] Παρουσιάζεται στο Παράρτημα Α

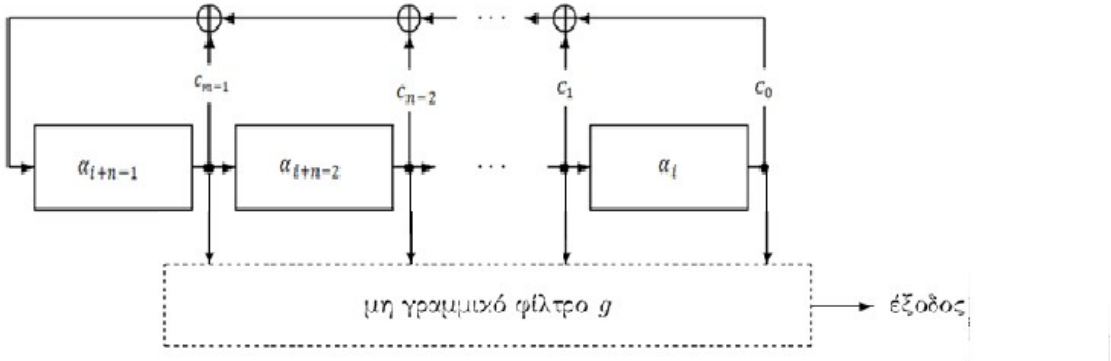
Συνοψίζοντας αν μια ακολουθία έχει γραμμική πολυπλοκότητα L, τότε αρκεί στον αντίπαλο να γνωρίζει $2L$ διαδοχικά bits της ακολουθίας για να μπορέσει να υπολογίσει το ελάχιστο LFSR που την παράγει και να ¨σπάσει¨ το σύστημα μας. Συνεπώς για να χρησιμοποιηθεί μια ακολουθία σε ένα κρυπτογραφικό σύστημα (πχ ως κλειδί σε έναν κρυπτογραφικό αλγόριθμο ροής) θα πρέπει η γραμμική πολυπλοκότητα της να είναι όσο το δυνατόν μεγαλύτερη.

3.7 Μη γραμμικές πράξεις και LFSRs

Οι πρωταρχικοί LFSRs παράγουν ακολουθίες με την μικρότερη δυνατή γραμμική πολυπλοκότητα (τις ακολουθίες μεγίστου μήκους). Θα ήταν αναμενόμενο λοιπόν -λόγο της μικρότερης δυνατής γραμμικής πολυπλοκότητας- να τους αποφεύγουμε. Όμως, λόγο των πολύ καλών -λοιπών- ιδιοτήτων τους συνεχίζουν να χρησιμοποιούνται ευρέως, για παράδειγμα ως δομικά συστατικά γεννήτριας κλειδοροής, κυρίως ως βασικά τμήματα ενός ευρύτερου μηχανισμού, που ουσιαστικά αποσκοπεί στο να αυξήσει τη γραμμική πολυπλοκότητα του συστήματος. Αυτό γίνεται, συνήθως, με την εισαγωγή μη-γραμμικών πράξεων που δρουν με κάποιο τρόπο στην κατά τα άλλα κανονική λειτουργία ενός LFSR.

Τετοιου ειδους μη-γραμμικες πράξεις κατηγοριοποιουνται ως εξής:

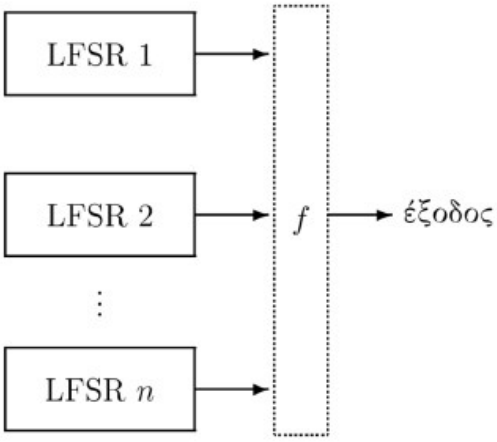
- Μη γραμμικά φίλτρα (non linear filter generators)



Σε αυτήν την περίπτωση εφαρμόζεται μια μη γραμμική λογική συνάρτηση g, το επονομαζόμενο μη γραμμικό φίλτρο (nonlinear filter) στις βαθμίδες ενός πρωταρχικού LFSR. Η έξοδος είναι στην πλειονότητα των περιπτώσεων μια ακολουθία με μεγάλη περίοδο και υψηλή γραμμική πολυπλοκότητα. Τα ιδανικά μη γραμμικά φίλτρα πρέπει να εξασφαλίζουν ομοιόμορφη κατανομή των 0 και 1 στην εξαγομένη ακολουθία. Ο ακριβής προσδιορισμός της

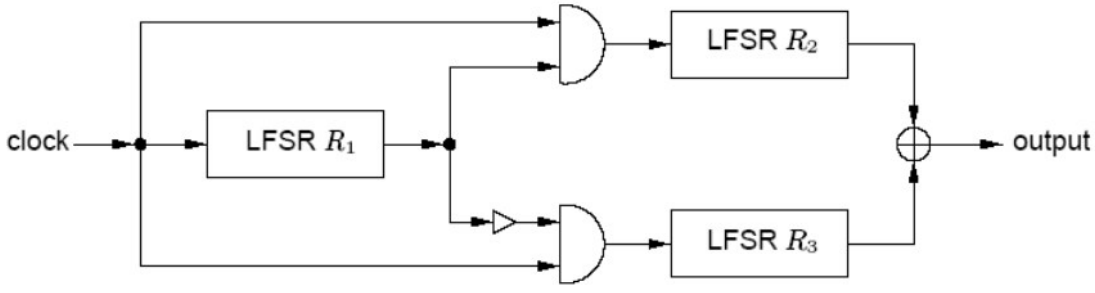
γραμμικής πολυπλοκότητας ακολουθιών που παράγονται από μη γραμμικά φίλτρα, αποτελεί ανοικτό ερευνητικό πρόβλημα ακόμα και σήμερα.

- Μη γραμμικοί συνδυαστές (nonlinear combination generators)



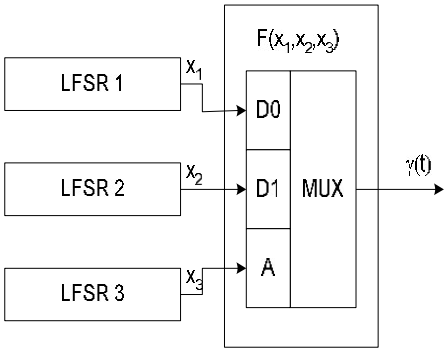
Σε αυτήν την περίπτωση συνδυάζονται πολλοί LFSRs έτσι ώστε οι έξοδοι τους να τροφοδοτούν μια μη γραμμική λογική συνάρτηση που ονομάζεται συνάρτηση-συνδυαστής (πχ ΓΨΑ Geffe). Και εδώ οι ιδανικοί μη γραμμικοί συνδυαστές πρέπει να εξασφαλίζουν ομοιόμορφη κατανομή των 0 και 1 στην εξαγομένη ακολουθία.

- Γεννήτριες ελεγχόμενες από ρολόι (Clock-controlled generators).



Σε αυτήν την περίπτωση το κάθε πότε αλλάζει η κατάσταση ενός LFSR, δηλαδή ο χρονισμός του εξαρτάται από την έξοδο ενός άλλου LFSR. Εκτός του ότι όλοι οι LFSR πρέπει να είναι πρωταρχικοί (όπως και στις προηγούμενες περιπτώσεις), θα πρέπει και το μέγεθος τους να είναι παραπλήσιο.

3.8 ΓΨΑ Geffe



Η ΓΨΑ Geffe ανήκει στην κατηγορία των μη γραμμικών συνδυαστών οι οποίοι χρησιμοποιούν (πρωταρχικούς κατά προτίμηση) LFSRs ως στοιχεία δομής. Χρησιμοποιεί τρείς το πλήθος LFSRs (LFSR1, LFSR2 και LFSR3), οι οποίοι παράγουν τις ακολουθίες χ_1, χ_2, χ_3, αντίστοιχα. Η Geffe εξασφαλίζει την ανάμιξη των ακολουθιών χ_1, χ_2, υπό τον έλεγχο της χ_3, με βάση τη συνάρτησης πολλαπλασιασμού

$$F(\chi_1, \chi_2, \chi_3) = \chi_1\chi_2 \oplus \chi_2\chi_3 \oplus \chi_3,$$

η οποία υλοποιείται με την βοήθεια του πολυπλέκτη $2 \rightarrow 1$, σύμφωνα με το παραπάνω σχήμα.

Όπως αναφέρθηκε προτιμούμε οι LFSRs να είναι πρωταρχικοί. Τότε, αν $L_i, i = 1,2,3$ το μήκος του κάθε LFSR αντίστοιχα, η κάθε μέγιστη περίοδος θα είναι ίση με $2^{L_i} - 1, i = 1,2,3$, αντίστοιχα.

Επιλέγουμε τα L_i έτσι ώστε να είναι πρώτα μεταξύ τους ανά 2.

Τότε η ακολουθία εξόδου γ(t) θα έχει:

- (μεγάλη) περίοδο ίση με $(2^{L_1} - 1) \cdot (2^{L_2} - 1) \cdot (2^{L_3} - 1)$ και
- γραμμική πολυπλοκότητα ίση με $L = L_1L_2 + L_2L_3 + L_3$

Όμως $P(\gamma(t) = \chi_1) = \frac{3}{4}$. Δηλαδή η πιθανότητα η έξοδος της Geffe να ταυτίζεται με την έξοδο του LFSR1 που χρησιμοποιείται ως είσοδος είναι 75%. Αυτό την κάνει ευάλωτη σε correlation κρυπτανάλυση. Η οποία βασίζεται στην ιδέα: αν η πιθανότητα να ταυτίζεται η έξοδος με κάποια από τις εξόδους των καταχωρητών, είναι μεγαλύτερη του $1/2$, τότε αν ένα ικανοποιητικά μεγάλο τμήμα της ακολουθίας εξόδου γίνει γνωστό, τότε μπορεί να βρεθεί η αρχική κατάσταση του συγκεκριμένου καταχωρητή.

Για το λόγο αυτό θα έπρεπε η συνάρτηση F να μην εμφανίζει (ικανή) στατιστική εξάρτηση ανάμεσα στην έξοδο και σε κάποιο υποσύνολο των εισόδων της, θα έπρεπε να είναι, δηλαδή, ανεπηρέαστη στη συσχέτιση.

3.9 Γραμμικές αναλογικές γεννήτριες (Linear Congruential Generators-LCGs)

Η μαθηματική φόρμουλα πάνω στην οποία βασίζεται η αρχιτεκτονική τους προτάθηκε πρώτη φορά από τον D.H.Lehmer. Έχουν μεγάλη εφαρμογή σε πειράματα προσομοίωσης και τυχαιοποιημένους αλγόριθμους. Δυστυχώς αν και είναι πολύ γρήγορες και παρότι περνούν με επιτυχία αρκετούς στατιστικούς δεν ενδείκνυται η χρήση τους σε κρυπτογραφικές εφαρμογές. Αυτό συμβαίνει γιατί αν αποκαλυφθεί ένα μέρος της παραγόμενης ακολουθίας, τότε μπορεί να υπολογιστούν από τον αντίπαλο και οι υπόλοιποι (επόμενοι) της όροι ακόμα και αν οι παράμετροι της είναι κρυφοί. Πρώτη φορά κατάφερε να τους 'σπάσει' ο Jim Reeds και στην συνέχεια η Joan Boyar.

Μια LCG δέχεται ως είσοδο τους μη αρνητικούς ακεραίους α, γ και n και δεχόμενη έναν μη αρνητικό ακέραιο Z_0 ως σπόρο, παράγει μια ακολουθία ακεραίων $Z_1, Z_2, \ldots$, βασιζόμενη στη σχέση:

$$Z_i = (\alpha Z_{i-1} + \gamma) mod n, i = 1,2,..$$

όπου ο α καλείται πολλαπλασιαστή, ο γ αύξηση και ο n διαιρέτης.

Επιπλέον πρέπει: $0 < n$, $\alpha < n$, $\gamma < n$ και $Z_0 < n$.

Προφανώς $0 \leq Z_i \leq n - 1,\ i = 1,2,\ldots$ και επομένως η γεννήτρια έχει μέγιστη περίοδο n.

Μια LCG δίνει μέγιστη μέγιστη (πλήρη) περίοδο n αν και μόνο αν οι ισχύουν οι ακόλουθες τρεις συνθήκες:

- Ο μόνος θετικός ακέραιος που διαιρεί ακριβώς το n και το γ είναι ο 1.
- Αν p πρώτος και $p \setminus n$, τότε $p \setminus (\alpha - 1)$.
- Αν $4 \setminus n$, τότε $4 \setminus (\alpha - 1)$.

Αν μια LCG πληροί τις τρεις προαναφερθείσες συνθήκες έχει μέγιστη περίοδο για κάθε (αποδεκτή) τιμή του σπόρου Z_0 . Αν όμως δεν έχει μέγιστη περίοδο, τότε το μήκος της περιόδου της εξαρτάται από το Z_0 και αναφερόμαστε σε αυτήν ως περίοδο της αρχικής τιμής της γεννήτριας.

Αν θέλουμε οι ψευδοτυχαίοι αριθμοί να ανήκουν στο διάστημα $[0,1]$ (πχ για μια εφαρμογή προσομοίωσης), προστίθεται ένα ακόμα βήμα:

$$U_i = {}^{Z_i}/_{n}, i = 1,2,\ldots$$

και έχουμε ως παραγόμενη την ακολουθία $U_1, U_2, \ldots$,

Ένα πρόβλημα που παρουσιάζεται όταν θέλουμε να πάρουμε ψευδοτυχαίους αριθμούς στο $[0,1]$ είναι ότι $U_i \in \{0, \frac{1}{n}, \frac{2}{n}, \ldots, \frac{n-1}{n}\}$, δηλαδή δεν θα πάρουμε ποτέ κάποιον ότι $U_i \in [\frac{0.2}{n}, \frac{0.7}{n}]$, παρότι θεωρητικώς θα έπρεπε να εμφανίζεται με πιθανότητα $\frac{0.5}{n} > 0$. Για να αντισταθμίζεται αυτό (ως έναν βαθμό), προτείνεται να επιλέγεται πολύ μεγάλος n, έτσι ώστε να έχουμε πολλές πιθανές τιμές για τα U_i, πχ για $n \geq 10^9$;έχουμε από ένα δισεκατομμύριο πιθανές τιμές και πάνω.

Όπως γίνεται αντιληπτό από την πρώτη συνθήκη μια LCG τείνει να συμπεριφέρεται διαφορετικά για $\gamma > 0$ (τότε ονομάζεται μεικτή LCG) και διαφορετικά για $\gamma = 0$ (τότε ονομάζεται πολλαπλασιαστική LCG).

3.10 ΓΨΑ Blum, Blum and Shub (BBS)

Η γεννήτρια BBS βασίζεται στο κρυπτοσύστημα του Blum. Το όνομα της αποτελείται από τα αρχικά των ερευνητών E.Blum, M.Blum και M.Shub. Ανήκει στις κρυπτογραφικές γεννήτριες και η κρυπτογραφική της συνάρτηση E_κ κατασκευάστηκε έχοντας ως βάση τις μονόδρομες συναρτήσεις. Το δυσεπίλυτο πρόβλημα της ανάλυσης ενός μεγάλου ακεραίου σε γινόμενο παραγόντων είναι η βάση της ασφάλειας της.

Περιγραφή:

1. Επιλέγουμε δυο μεγάλους πρώτους αριθμούς p, q ίδιου μεγέθους, τέτοιους ώστε $p \equiv 3 mod 4$ και $q \equiv 3 mod 4$, τότε $n = pq$ ο ακέραιος αριθμός του Blum.
2. Έστω θετικός ακέραιος x τέτοιος ώστε: $\mu\kappa\delta(x, n) = 1$. Παίρνουμε για αρχική τιμή (σπόρο) $x_0 = x^2 mod n$,.
3. Τότε $x_\iota = x_{\iota-1}^2 mod n$, $i = 1,2,..,m-1$
4. Για κάθε $i = 0,1,..,m-1$ θέτουμε b_i = το μικρότερο σε αξία bit του αριθμού x_ι, δηλαδή το δεξιοτερο στοιχείο (bit) της δυαδικής αναπαράστασης του αριθμού x_ι.

Τότε παραγόμενη ακολουθία είναι η $BBS_{n,m}(x_0) = b_0 b_1 \dots b_{m-1}$

Σχόλια:

Ο αριθμός x αποτελεί το κλειδί της γεννήτριας, οι p, q είναι μυστικοί, ενώ το n δημοσιοποιείται, έτσι ώστε να μπορεί να χρησιμοποιήσει τη γεννήτρια οποιοσδήποτε και να παράγει bits. Χωρίς όμως να είναι δυνατόν από κάποιον κρυπταναλυτή (εκτός αν καταφέρει να παραγοντοποιήσει το n) να προβλέψει το αποτέλεσμα της γεννήτρια, ούτε καν να κάνει μια ¨καλή πρόβλεψη¨ για αυτήν. Αφού είναι δεξιά και αριστερά μη προβλέψιμη. Δοθέντος, δηλαδή, κάποιου τμήματος της ακολουθίας, ο αντίπαλος δεν μπορεί να προβλέψει ούτε το επόμενο, αλλά, ούτε και το προηγούμενο στοιχείο (αυτού του τμήματος). Και αυτό επιτυγχάνεται όχι με την χρήση κάποιου εξεζητημένου δικτύου ιδιάζουσας αρχιτεκτονικής, αλλά με τη χρήση ενός βασικού μαθηματικού προβλήματος. Το βασικό μειονέκτημα της γεννήτριας είναι η ταχύτητα της (είναι αργή). Γεγονός που κάνει δύσκολη την ενσωμάτωση της σε κρυπτογραφικούς αλγόριθμους ροής (απαιτούν ταχύτητα). Όμως η ασφάλεια της την

καθιστά από τις βασικές επιλογές σε ζητήματα υψηλής ασφαλείας, όπως στην δημιουργία κλειδιών.

Η πιο σημαντική ιδιότητα της BBS είναι ότι αν κάποιος γνωρίζει τα p,q, για να υπολογίσει κάποιο στοιχείο της ακολουθίας, έστω το b_i, δεν χρειάζεται να υπολογίσει πρώτα όλα τα προηγούμενα, αλλά χρησιμοποιώντας τον τύπο: $x_i = x_0^{(2i) mod((p-1)(q-1))}$, βρίσκει το x_i και από αυτό το b_i. Γεγονός που δίνει τη δυνατότητα να χρησιμοποιηθεί, αυτή η ¨κρυπτογραφικά δυνατή¨ ΓΨΑ ακόμα και σε ένα κρυπτοσύστημα το οποίο δίνει πρόσβαση σε έναν τυχαίο φάκελο.

Η ασφάλεια της BBS ανάγεται στην κλασσική θεωρία αριθμών: Έστω $\widetilde{\mathbb{Z}_n^*} = \{z \in \mathbb{Z} / 0 < z < n,\ z \nmid p \ και \ z \nmid q\}$ και QR_n το υποσύνολο του $\widetilde{\mathbb{Z}_n^*}$ που αποτελείται από τα τετραγωνικά υπόλοιπα του n. Το πλήθος των στοιχείων του $\widetilde{\mathbb{Z}_n^*}$ ισούται με $(p-1)(q-1)$, με το ένα τέταρτο των οποίων να αποτελεί τα στοιχεία του QR_n. Κάθε στοιχείο του QR_n (κάθε τετραγωνικό υπόλοιπο) έχει τέσσερις διακεκριμένες τετραγωνικές ρίζες στο $\widetilde{\mathbb{Z}_n^*}$, από τις οποίες όμως μόνο μια ανήκει και στο QR_n (δηλαδή μόνο μια από τις τέσσερις τετραγωνικές ρίζες ενός τετραγωνικού υπολοίπου είναι επίσης τετραγωνικό υπόλοιπο)., αυτή θα την ονομάζουμε πρωταρχική τετραγωνική ρίζα..
Το πρόβλημα ανεύρεσης τετραγωνικών ριζών $modulo\ n$ είναι τόσο δύσκολο όσο και αυτό της παραγοντοποίησης του n. Αν δεν είναι γνωστή η παραγοντοποίηση του n , τότε δεν υπάρχει γνωστή αποδοτική διαδικασία για την επίλυση του προβλήματος των τετραγωνικών υπολοίπων $modulo\ n$.

Παράδειγμα[6]

Έστω p=24672462467892469787(=6168115616973117446·4+3)

και q=396736894567834589803(=991842236419586474 50·4+3), δυο πρώτοι αριθμοί οι οποίοι όταν διαιρεθούν με το 4 δίνουν υπόλοιπο 3.

Θέτουμε n=pq=9788476140853110794168855217413715781961

Επιλέγουμε χ=873245647888478349013 καθώς πρέπει μκδ(χ,n)=1

Η αρχική τιμή χ_0 είναι $x_0 = x^2 mod n$=884529871047878009708991774601012286317 2

Τότε οι τιμές για τους $\chi_1, \chi_2, \dots, \chi_{13}$, είναι:

χ_1=711889428113132952274596245549812382240 8

χ_2=314517460888889316415138015206070451822 7

χ_3=489800778230715623327223318557489943035 5

χ_4=393545781893511292234709354618967231038 9

χ_5=675099511510097048901761303198740246040

[6] Β.Α ΚΑΤΟΣ, Γ.Χ ΣΤΕΦΑΝΙΔΗΣ, ΤΕΧΝΙΚΕΣ ΚΡΥΠΤΟΓΡΑΦΙΑΣ ΚΑΙ ΚΡΥΠΤΑΝΑΛΥΣΗΣ, ΕΚΔΟΣΕΙΣ ΖΥΓΟΣ

χ_6=428991482877174013354619065826651517132 6

χ_7=443106671145437826089038638559381752166 8

χ_8=733687612419504639741423533367500537243 6

χ_9=445071407076962646306299549503965809406

χ_{10}=297452148493377896922269953037509550661 0

χ_{11}=411584510969558468488918520236621988196 6

χ_{12}=246100424258716107455505690264137829302 6

χ_{13}=534148949234149580093205789114991579817 3

Λαμβάνοντας το λιγότερο σημαντικό bit για κάθε χ_i, με τον έλεγχο αν ο ακέραιος είναι άρτιος ή περιττός, παράγεται η ακολουθία:

$BBS_{n,14}(x_0) = b_0 b_1 \dots b_{13}$=00111000000001

3.11 ΓΨΑ Sha-1

Η ΓΨΑ Sha-1, βασίζεται στον hash αλγόριθμο Sha-1. Δέχεται ως είσοδο μια ακολουθία t μήκους 160 bits και μια ακολουθία c μήκους b bits, όπου $160 \leq b \leq 512$ και δίνει ως έξοδο μια ακολουθία μήκους 160 bits, η οποία είναι αποτέλεσμα μιας μονόδρομης συνάρτησης $G(t,c)$. Λόγω του μεγέθους σε bits που χρησιμοποιεί στις εισόδους, στις εξόδους και στα τμήματα στα οποία χωρίζει τις ακολουθίες του χρησιμοποιεί το δεκαεξαδικό σύστημα αρίθμησης στον κώδικα του για ευκολία. Έτσι κάθε ακολουθία τεσσάρων bits αντιστοιχεί σε ένα δεκαεξαδικό ψηφίο σύμφωνα με τον πίνακα:

0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	a
1011	b
1100	c
1101	d
1110	e
1111	f

Συμβολικά $1110\,0001_2 = e1_{16}$. Άρα κάθε block μεγέθους 32 bits (που χρησιμοποιείται), θα αντιστοιχεί σε μια ακολουθία από 8 δεκαεξαδικά ψηφία. Τα δυο δεξιότερα δεκαεξαδικά στοιχεία μιας ακολουθίας μας δίνουν το λιγότερο σημαντικό byte. Η Sha-1 ως γεννήτρια παρουσιάστηκε πρώτη φορά στις 27/1/2000 με το FIPS (Federal Information Processing Standard) pub 186-2 στο οποίο ορίζεται ως πρότυπο. Ο αλγόριθμος κατακερματισμού (hash) Sha-1 (FIPS pub 180-1, 17/4/1995) στον οποίο βασίζεται, αποτελεί μια εξελιγμένη έκδοση της οικογένειας των Sha αλγορίθμων που εισήχθηκαν με την έκδοση SHS (Hash Secure Standard) (FIPS pub 180, 11/5/93), στις ΗΠΑ από το Εθνικό Ινστιτούτο Πρότυπων και Τεχνολογίας σε συνεργασία (NIST) με την Εθνική Υπηρεσία Ασφαλείας (NSA).

Διαδικασία:

1. Η 160-bit c χωρίζεται σε 5 blocks $H_i, i = 1, \dots, 5$, μεγέθους 32 bits το καθένα, δηλαδή $t = H_1 \parallel H_2 \parallel H_3 \parallel H_4 \parallel H_5 \parallel$
2. $512 - b$ το πλήθος 0 συμπληρώνονται στο τέλος της c, έτσι ώστε να προκύψει μια ακολουθία/μήνυμα X μήκους 512 bits, δηλαδή $X = c \parallel 0^{512-b}$
3. Διαιρείται η X σε 16 τμήματα/λέξεις $x_i, i = 0,1, \dots, 15$, μεγέθους 32 bits η κάθε μια.
4. Χρησιμοποιώντας ως βάση τις 16 λέξεις σχηματίζονται άλλες 64 λέξεις για να φτάσουμε σε ένα σύνολο 80 λέξεων, σύμφωνα με το κάτωθι: Για j=16 έως 79
   $x_j = \left(\left(x_{j-3} \oplus x_{j-8} \oplus x_{j-14} \oplus x_{j-16} \right) \leftarrow 1 \right)$
5. Αρχικοποιούνται οι τιμές των 32-bit μεταβλητών A, B, C, D, E ως: $(A, B, C, D, E) = (H_1, H_2, H_3, H_4, H_5)$
6. Ακολουθούν 4 γύροι, 20 επαναλήψεων ο καθένας, που έχουν ως σκοπό να ¨ανακατέψουν/.αλλάξουν¨ τις τιμές των A, B, C, D, E, που παρότι εισάγονται επιπλέον μόνο οι προκαθορισμένες τιμές y_i στην διαδικασία, σε τέτοιο βαθμό έτσι ώστε φαινομενικά να μην σχετίζονται με τις αρχικές τους τιμές.

   ($1^{ος}$ γύρος) Για j=0 έως 19
   $t \leftarrow \left((A \leftarrow 5) + f(B, C, D) + E + x_j + y_1 \right)$,
   $(A, B.C.D.E) \leftarrow (t, A, B \leftarrow 30, C, D)$

   ($2^{ος}$ γύρος) Για j=20 έως 39
   $t \leftarrow \left((A \leftarrow 5) + h(B, C, D) + E + x_j + y_2 \right)$,
   $(A, B.C.D.E) \leftarrow (t, A, B \leftarrow 30, C, D)$

   ($3^{ος}$ γύρος) Για j=40 έως 59

$$t \leftarrow \left((A \hookleftarrow 5) + g(B,C,D) + E + x_j + y_3\right),$$
$$(A,B.C.D.E) \leftarrow (t, A, B \hookleftarrow 30, C, D)$$

($4^{ος}$ γύρος) Για j=60 έως 79

$$t \leftarrow \left((A \hookleftarrow 5) + h(B,C,D) + E + x_j + y_4\right),$$
$$(A,B.C.D.E) \leftarrow (t, A, B \hookleftarrow 30, C, D)$$

7. Υπολογίζονται οι νέες τιμές των $H_i, i = 1, \ldots, 5$ σύμφωνα με:
$(H_1, H_2, H_3, H_4, H_5) = (H_1 + A, H_2 + B, H_3 + C, H_4 + D, H_5 + E)$
8. Δίνεται η ακολουθία εξόδου $G(t,c) = H_1 \parallel H_2 \parallel H_3 \parallel H_4 \parallel H_5 \parallel$

Ορισμοί και επεξηγήσεις:

y_1	$5a827999_{16}$
y_2	$6ed9eba1_{16}$
y_3	$8f1bbcdc_{16}$
y_4	$ca62c1d6_{16}$
+	η πρόσθεση $modulo 2^{32}$
A, B, C, D, E	μεταβλητές που αναπαριστούν ποσότητες μεγέθους 16 bits η κάθεμια
$\bar{A}$	Το συμπλήρωμα το Α (σε επίπεδο bit)
$A \hookleftarrow n$	Κυκλική μετατόπιση του Α προς τα αριστερά κατά n bit
AB	A ΚΑΙ B(σε επίπεδο bit)
$A \vee B$	A Ή B (σε επίπεδο bit)
$f(B,C,D)$	$BC \bigvee \bar{B}D$
$g(B,C,D)$	$BC \bigvee BD \bigvee CD$
$h(B,C,D)$	$B \oplus C \oplus D$

Ασφάλεια:

Η ασφάλεια του Sha-1 βασίζεται στην πολύπλοκη αρχιτεκτονική του. Κατά τη λειτουργία του συναντώνται αλλεπάλληλα δίκτυα, μέσα στα οποία υπάρχει αλληλουχία διαδικασιών. Όπως οι μη γραμμικές συναρτήσεις f και g (η μη γραμμικότητα αναβαθμίζει σημαντικά την ασφάλεια), η γραμμική συνάρτηση h και μια πληθώρα εκλεπτυσμένων κυκλικών μετατοπίσεων (διαφορετικού πλήθους bits κάθε φορά). Ο συνδυασμός των οποίων παίρνει τα αρχικά δεδομένα (και κάποιες σταθερές), δημιουργεί καινούργια χρησιμοποιώντας τα, στη συνέχεια τα επεξεργάζεται/ανακατεύει σε τέτοιο βαθμό (μέσω των δικτύων του), που στο τέλος δίνοντας ως έξοδο μόνο κάποια κομμάτια όλων αυτών (η εξαγόμενη ακολουθία έχει μέγεθος σημαντικά μικρότερο των δεδομένων εισόδου), η τελική ακολουθία να φαίνεται τυχαία και να μην μπορεί να συνδυαστεί από κάποιων με το αρχικό μήνυμα/είσοδο, παρότι προέκυψε από αυτό. Βλέπουμε, δηλαδή, μια εντελώς διαφορετική φιλοσοφία στην οποία βασίζεται η ασφάλεια, από ότι πχ στην BBS, της οποίας η ασφάλεια βασιζόταν σε ένα δυσεπίλυτο μαθηματικό πρόβλημα.

4 Αξιολόγηση και συμπεράσματα

Εισαγωγή

Στο παρόν κεφάλαιο ελέγχονται και αξιολογούνται οι ΓΨΑ που παρουσιάστηκαν στο κεφάλαιο 3 μέσω της σουίτας στατιστικών ελέγχων που παρουσιάστηκε στο κεφάλαιο 2, με σκοπό την εξαγωγή συνολικών συμπερασμάτων. Πληθώρα ακολουθιών που εξετάζονται από την σουίτα, οι οποίες σκοπίμως ποικίλουν σε μέγεθος. Έτσι ώστε να ελεγχθούν ακολουθίες που για τους κατασκευαστές της θεωρούνται μικρού, μεσαίου και μεγάλου μεγέθους. Γίνεται και μια ανάλυση της συμπεριφοράς της ίδιας της σουίτας ως προς την ευαισθησία που παρουσιάζει στις εισόδους της. Διευκρινήσεις σχετικά με τη δημιουργία των ακολουθιών, πέραν του κεφαλαίου 3, μπορούν να αντληθούν από το Παράρτημα Β′. Η δημιουργία των ακολουθιών και η εκτέλεση των ελέγχων έγιναν σε macOS Mojave 10.14.3

4.1 Αξιολόγηση ΓΨΑ πρωταρχικού LFSR μεγέθους 20-bit

Με αρχική κατάσταση 11000111000111111111 και χαρακτηριστικό πολυώνυμο $f(x) = x^{20} + x^3 + 1$, δημιουργήθηκε, παίρνοντας ως έξοδο το λιγότερο σημαντικό bit κάθε κατάστασης, ακολουθία μεγίστου μήκους $2^{20} - 1 = 1048575$ bits, η οποία αποτελείται από 524288 το πλήθος άσσους και 524287 το πλήθος μηδενικά. Η ακολουθίας αυτή εξετάστηκε από τη σουίτα στατιστικών ελέγχων που παρουσιάστηκε στο Κεφάλαιο 2, δίνοντας μας τα εξής αποτελέσματα:

Τεστ	P-value
Frequency (Monobit)	0.999221 (ΕΠΙΤΥΧΙΑ)
Frequency within a block ($M = 128$)	0.061212 (ΕΠΙΤΥΧΙΑ)
Runs	0.997662 (ΕΠΙΤΥΧΙΑ)
Longest Run of Ones in a block ($M = 10^4, N = 104$)	0.945406 (ΕΠΙΤΥΧΙΑ)
Binary Matrix Rank	0.000000 (ΑΠΟΤΥΧΙΑ)
DFT (Spectral)	0.000000 (ΑΠΟΤΥΧΙΑ)
Non-Overlapping Template Matching	Για m=9 υπολογίζονται 148 P-values δείτε αναλυτικά παρακάτω
Overlapping Template Matching ($m = 9$)	0.057312 (ΕΠΙΤΥΧΙΑ)
Maurer's "Universal Statistical" ($L = 7, Q = 1280, K = 148516$)	0.208616 (ΕΠΙΤΥΧΙΑ)
Linear Complexity (M=500)	0.000000 (ΑΠΟΤΥΧΙΑ)
Serial ($m = 16$) P-value1	1.000000 (ΕΠΙΤΥΧΙΑ)
Serial ($m = 16$) P-value2	1.000000 (ΕΠΙΤΥΧΙΑ)
Approximate Entropy ($m = 10$)	1.000000 (ΕΠΙΤΥΧΙΑ)
Cumulative Sums (Cusum) forward	0.599142 (ΕΠΙΤΥΧΙΑ)
Cumulative Sums (Cusum) reverse	0.600018 (ΕΠΙΤΥΧΙΑ)
Random Excursions	Υπολογίζονται 8 P-values δείτε αναλυτικά παρακάτω
Random Excursions Variant	Υπολογίζονται 18 P-values δείτε αναλυτικά παρακάτω

Για το Non-overlapping Template Matching έχουμε για τα μη περιοδικά μοτίβο που εξετάστηκαν:

Μοτίβο (B) P-value Αποτέλεσμα Α/α

000000001 0.636380 ΕΠΙΤΥΧΙΑ 0

000000011 0.737769 ΕΠΙΤΥΧΙΑ 1

000000101 0.522095 ΕΠΙΤΥΧΙΑ 2

000000111 0.226801 ΕΠΙΤΥΧΙΑ 3

000001001 0.025563 ΕΠΙΤΥΧΙΑ 4

000001011 0.325370 ΕΠΙΤΥΧΙΑ 5

000001101 0.747812 ΕΠΙΤΥΧΙΑ 6

000001111 0.180169 ΕΠΙΤΥΧΙΑ 7

000010001 0.511780 ΕΠΙΤΥΧΙΑ 8
000010011 0.792946 ΕΠΙΤΥΧΙΑ 9
000010101 0.930886 ΕΠΙΤΥΧΙΑ 10
000010111 0.562329 ΕΠΙΤΥΧΙΑ 11
000011001 0.531621 ΕΠΙΤΥΧΙΑ 12
000011011 0.993684 ΕΠΙΤΥΧΙΑ 13
000011101 0.628675 ΕΠΙΤΥΧΙΑ 14
000011111 0.665371 ΕΠΙΤΥΧΙΑ 15
000100011 0.631389 ΕΠΙΤΥΧΙΑ 16
000100101 0.417065 ΕΠΙΤΥΧΙΑ 17
000100111 0.700907 ΕΠΙΤΥΧΙΑ 18
000101001 0.628675 ΕΠΙΤΥΧΙΑ 19
000101011 0.872793 ΕΠΙΤΥΧΙΑ 20
000101101 0.765958 ΕΠΙΤΥΧΙΑ 21
000101111 0.042610 ΕΠΙΤΥΧΙΑ 22
000110011 0.077546 ΕΠΙΤΥΧΙΑ 23
000110101 0.921740 ΕΠΙΤΥΧΙΑ 24
000110111 0.890964 ΕΠΙΤΥΧΙΑ 25
000111001 0.411679 ΕΠΙΤΥΧΙΑ 26
000111011 0.931924 ΕΠΙΤΥΧΙΑ 27
000111101 0.631389 ΕΠΙΤΥΧΙΑ 28
000111111 0.291801 ΕΠΙΤΥΧΙΑ 29
001000011 0.184575 ΕΠΙΤΥΧΙΑ 30
001000101 0.613322 ΕΠΙΤΥΧΙΑ 31
001000111 0.483030 ΕΠΙΤΥΧΙΑ 32
001001011 0.123083 ΕΠΙΤΥΧΙΑ 33
001001101 0.711635 ΕΠΙΤΥΧΙΑ 34
001001111 0.641342 ΕΠΙΤΥΧΙΑ 35
001010011 0.197270 ΕΠΙΤΥΧΙΑ 36
001010101 0.139020 ΕΠΙΤΥΧΙΑ 37
001010111 0.539462 ΕΠΙΤΥΧΙΑ 38
001011011 0.753027 ΕΠΙΤΥΧΙΑ 39
001011101 0.837683 ΕΠΙΤΥΧΙΑ 40
001011111 0.499006 ΕΠΙΤΥΧΙΑ 41
001100101 0.153741 ΕΠΙΤΥΧΙΑ 42
001100111 0.829186 ΕΠΙΤΥΧΙΑ 43
001101011 0.384640 ΕΠΙΤΥΧΙΑ 44

001101101 0.826842 ΕΠΙΤΥΧΙΑ 45
001101111 0.984175 ΕΠΙΤΥΧΙΑ 46
001110101 0.939445 ΕΠΙΤΥΧΙΑ 47
001110111 0.251399 ΕΠΙΤΥΧΙΑ 48
001111011 0.583681 ΕΠΙΤΥΧΙΑ 49
001111101 0.675714 ΕΠΙΤΥΧΙΑ 50
001111111 0.657666 ΕΠΙΤΥΧΙΑ 51
010000011 0.785880 ΕΠΙΤΥΧΙΑ 52
010000111 0.258558 ΕΠΙΤΥΧΙΑ 53
010001011 0.422491 ΕΠΙΤΥΧΙΑ 54
010001111 0.708065 ΕΠΙΤΥΧΙΑ 55
010010011 0.552618 ΕΠΙΤΥΧΙΑ 56
010010111 0.592634 ΕΠΙΤΥΧΙΑ 57
010011011 0.822912 ΕΠΙΤΥΧΙΑ 58
010011111 0.308566 ΕΠΙΤΥΧΙΑ 59
010100011 0.887692 ΕΠΙΤΥΧΙΑ 60
010100111 0.379513 ΕΠΙΤΥΧΙΑ 61
010101011 0.946954 ΕΠΙΤΥΧΙΑ 62
010101111 0.313670 ΕΠΙΤΥΧΙΑ 63
010110011 0.325370 ΕΠΙΤΥΧΙΑ 64
010110111 0.213525 ΕΠΙΤΥΧΙΑ 65
010111011 0.276254 ΕΠΙΤΥΧΙΑ 66
010111111 0.203647 ΕΠΙΤΥΧΙΑ 67
011000111 0.881019 ΕΠΙΤΥΧΙΑ 68
011001111 0.717869 ΕΠΙΤΥΧΙΑ 69
011010111 0.393529 ΕΠΙΤΥΧΙΑ 70
011011111 0.889660 ΕΠΙΤΥΧΙΑ 71
011101111 0.938958 ΕΠΙΤΥΧΙΑ 72
011111111 0.205516 ΕΠΙΤΥΧΙΑ 73
100000000 0.636849 ΕΠΙΤΥΧΙΑ 74
100010000 0.739070 ΕΠΙΤΥΧΙΑ 75
100100000 0.255235 ΕΠΙΤΥΧΙΑ 76
100101000 0.822912 ΕΠΙΤΥΧΙΑ 77
100110000 0.892260 ΕΠΙΤΥΧΙΑ 78
100111000 0.499006 ΕΠΙΤΥΧΙΑ 79
101000000 0.747812 ΕΠΙΤΥΧΙΑ 80
101000100 0.957605 ΕΠΙΤΥΧΙΑ 81

101001000 0.280965 ΕΠΙΤΥΧΙΑ 82
101001100 0.615126 ΕΠΙΤΥΧΙΑ 83
101010000 0.282746 ΕΠΙΤΥΧΙΑ 84
101010100 0.637722 ΕΠΙΤΥΧΙΑ 85
101011000 0.653109 ΕΠΙΤΥΧΙΑ 86
101011100 0.586364 ΕΠΙΤΥΧΙΑ 87
101100000 0.732035 ΕΠΙΤΥΧΙΑ 88
101100100 0.977757 ΕΠΙΤΥΧΙΑ 89
101101000 0.633198 ΕΠΙΤΥΧΙΑ 90
101101100 0.892260 ΕΠΙΤΥΧΙΑ 91
101110000 0.443792 ΕΠΙΤΥΧΙΑ 92
101110100 0.566760 ΕΠΙΤΥΧΙΑ 93
101111000 0.906657 ΕΠΙΤΥΧΙΑ 94
101111100 0.768112 ΕΠΙΤΥΧΙΑ 95
110000000 0.776210 ΕΠΙΤΥΧΙΑ 96
110000010 0.499853 ΕΠΙΤΥΧΙΑ 97
110000100 0.538589 ΕΠΙΤΥΧΙΑ 98
110001000 0.342119 ΕΠΙΤΥΧΙΑ 99
110001010 0.280965 ΕΠΙΤΥΧΙΑ 100
110010000 0.661254 ΕΠΙΤΥΧΙΑ 101
110010010 0.444592 ΕΠΙΤΥΧΙΑ 102
110010100 0.849012 ΕΠΙΤΥΧΙΑ 103
110011000 0.080196 ΕΠΙΤΥΧΙΑ 104
110011010 0.340758 ΕΠΙΤΥΧΙΑ 105
110100000 0.889660 ΕΠΙΤΥΧΙΑ 106
110100010 0.699114 ΕΠΙΤΥΧΙΑ 107
110100100 0.269874 ΕΠΙΤΥΧΙΑ 108
110101000 0.389069 ΕΠΙΤΥΧΙΑ 109
110101010 0.516068 ΕΠΙΤΥΧΙΑ 110
110101100 0.858575 ΕΠΙΤΥΧΙΑ 111
110110000 0.779562 ΕΠΙΤΥΧΙΑ 112
110110010 0.899891 ΕΠΙΤΥΧΙΑ 113
110110100 0.485537 ΕΠΙΤΥΧΙΑ 114
110111000 0.664871 ΕΠΙΤΥΧΙΑ 115
110111010 0.741699 ΕΠΙΤΥΧΙΑ 116
110111100 0.679324 ΕΠΙΤΥΧΙΑ 117
111000000 0.151212 ΕΠΙΤΥΧΙΑ 118

111000010 0.392783 ΕΠΙΤΥΧΙΑ 119
111000100 0.885043 ΕΠΙΤΥΧΙΑ 120
111000110 0.642247 ΕΠΙΤΥΧΙΑ 121
111001000 0.865049 ΕΠΙΤΥΧΙΑ 122
111001010 0.887692 ΕΠΙΤΥΧΙΑ 123
111001100 0.085304 ΕΠΙΤΥΧΙΑ 124
111010000 0.419385 ΕΠΙΤΥΧΙΑ 125
111010010 0.781248 ΕΠΙΤΥΧΙΑ 126
111010100 0.385376 ΕΠΙΤΥΧΙΑ 127
111010110 0.244392 ΕΠΙΤΥΧΙΑ 128
111011000 0.700011 ΕΠΙΤΥΧΙΑ 129
111011010 0.477205 ΕΠΙΤΥΧΙΑ 130
111011100 0.923944 ΕΠΙΤΥΧΙΑ 131
111100000 0.362999 ΕΠΙΤΥΧΙΑ 132
111100010 0.998525 ΕΠΙΤΥΧΙΑ 133
111100100 0.787955 ΕΠΙΤΥΧΙΑ 134
111100110 0.641342 ΕΠΙΤΥΧΙΑ 135
111101000 0.207835 ΕΠΙΤΥΧΙΑ 136
111101010 0.267581 ΕΠΙΤΥΧΙΑ 137
111101100 0.593531 ΕΠΙΤΥΧΙΑ 138
111101110 0.672102 ΕΠΙΤΥΧΙΑ 139
111110000 0.057714 ΕΠΙΤΥΧΙΑ 140
111110010 0.947406 ΕΠΙΤΥΧΙΑ 141
111110100 0.468120 ΕΠΙΤΥΧΙΑ 142
111110110 0.933975 ΕΠΙΤΥΧΙΑ 143
111111000 0.121138 ΕΠΙΤΥΧΙΑ 144
111111010 0.535972 ΕΠΙΤΥΧΙΑ 145
111111100 0.244135 ΕΠΙΤΥΧΙΑ 146
111111110 0.198857 ΕΠΙΤΥΧΙΑ 147

Για το Random Excursions test (μετά από 1281 κύκλους J) έχουμε:

ΑΠΟΤΥΧΙΑ x = -4 P-value = 0.001067
ΕΠΙΤΥΧΙΑ x = -3 P-value = 0.020360
ΕΠΙΤΥΧΙΑ x = -2 P-value = 0.052302
ΕΠΙΤΥΧΙΑ x = -1 P-value = 0.056058
ΕΠΙΤΥΧΙΑ x = 1 P-value = 0.285743
ΕΠΙΤΥΧΙΑ x = 2 P-value = 0.555148

ΕΠΙΤΥΧΙΑ x = 3 P-value = 0.083651

ΕΠΙΤΥΧΙΑ x = 4 P-value = 0.116131

Για το Random Excursions Variant test (μετά από 1281 κύκλους J) έχουμε:

ΕΠΙΤΥΧΙΑ (x = -9) P-value = 0.142581

ΕΠΙΤΥΧΙΑ (x = -8) P-value = 0.128478

ΕΠΙΤΥΧΙΑ (x = -7) P-value = 0.149556

ΕΠΙΤΥΧΙΑ (x = -6) P-value = 0.117198

ΕΠΙΤΥΧΙΑ (x = -5) P-value = 0.047453

ΕΠΙΤΥΧΙΑ (x = -4) P-value = 0.014315

ΑΠΟΤΥΧΙΑ (x = -3) P-value = 0.006163

ΕΠΙΤΥΧΙΑ (x = -2) P-value = 0.030218

ΕΠΙΤΥΧΙΑ (x = -1) P-value = 0.286039

ΕΠΙΤΥΧΙΑ (x = 1) P-value = 0.220611

ΕΠΙΤΥΧΙΑ (x = 2) P-value = 0.197424

ΕΠΙΤΥΧΙΑ (x = 3) P-value = 0.344462

ΕΠΙΤΥΧΙΑ (x = 4) P-value = 0.540329

ΕΠΙΤΥΧΙΑ (x = 5) P-value = 0.812596

ΕΠΙΤΥΧΙΑ (x = 6) P-value = 0.947756

ΕΠΙΤΥΧΙΑ (x = 7) P-value = 0.886712

ΕΠΙΤΥΧΙΑ (x = 8) P-value = 0.947127

ΕΠΙΤΥΧΙΑ (x = 9) P-value = 0.662807

Όπως βλέπουμε μέσα από αναλυτική παρουσίαση η ακολουθία που παράχθηκε αποτυγχάνει πλήρως στα tests: Binary Matrix Rank, DFT και Linear Complexity. Και εμφανίζει από μια P-value που αποτυγχάνει στα τεστ Random Excursions και Random Excursions Variant. Για την καλύτερη αξιολόγηση της γεννήτριας υποβλήθηκαν σε έλεγχο 100 ακολουθίες μήκους 10480 bits η κάθε μια, που δημιουργήθηκαν από αυτή. Τα συνοπτικά αποτελέσματα όπως τα εμφανίζει η σουίτα είναι:

RESULTS FOR THE UNIFORMITY OF P-VALUES AND THE PROPORTION OF PASSING SEQUENCES

generator is <LFSR_20_3_last_row_final.txt>

C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	P-VALUE		PROPORTION		STATISTICAL TEST
7	9	6	8	18	5	19	14	8	6	0.004981		99/100		Frequency
17	6	7	7	2	5	7	7	9	33	0.000000	*	91/100	*	BlockFrequency
6	12	7	8	7	12	15	8	8	17	0.171867		99/100		CumulativeSums
6	7	7	9	7	9	8	15	14	18	0.080519		99/100		CumulativeSums
6	10	8	14	5	12	12	11	9	13	0.534146		98/100		Runs
14	10	6	7	10	12	12	13	5	11	0.494392		99/100		LongestRun
100	0	0	0	0	0	0	0	0	0	0.000000	*	0/100	*	Rank
34	11	7	4	7	12	4	10	6	5	0.000000	*	83/100	*	FFT
10	6	16	6	6	13	15	10	7	11	0.171867		97/100		NonOverlappingTemplate
13	12	11	4	11	16	11	10	3	9	0.129620		97/100		NonOverlappingTemplate
18	16	10	7	8	7	11	8	6	9	0.108791		95/100	*	NonOverlappingTemplate
29	9	5	9	10	9	9	6	4	10	0.000001	*	93/100	*	NonOverlappingTemplate
16	10	6	8	13	17	8	7	5	10	0.085587		95/100	*	NonOverlappingTemplate
8	7	9	9	8	13	12	9	10	15	0.759756		99/100		NonOverlappingTemplate
7	10	8	9	5	15	15	9	10	12	0.401199		98/100		NonOverlappingTemplate
12	9	10	10	6	15	14	11	4	9	0.350485		96/100		NonOverlappingTemplate
8	7	10	5	14	13	12	13	9	9	0.554420		96/100		NonOverlappingTemplate
7	7	12	3	14	12	12	8	11	14	0.236810		100/100		NonOverlappingTemplate
12	6	10	14	7	8	9	13	6	15	0.350485		98/100		NonOverlappingTemplate
13	11	6	5	11	10	9	16	9	10	0.437274		96/100		NonOverlappingTemplate
9	11	8	7	7	15	16	11	6	10	0.334538		97/100		NonOverlappingTemplate
13	8	10	8	8	13	16	6	6	12	0.334538		97/100		NonOverlappingTemplate
5	10	6	8	5	16	15	12	10	13	0.108791		100/100		NonOverlappingTemplate
12	7	7	4	14	16	11	12	7	10	0.191687		98/100		NonOverlappingTemplate
5	13	10	7	11	4	11	9	13	17	0.122325		100/100		NonOverlappingTemplate
12	9	4	9	11	13	12	11	8	11	0.719747		97/100		NonOverlappingTemplate
7	7	8	8	10	15	17	7	9	12	0.249284		98/100		NonOverlappingTemplate
9	4	5	6	18	11	13	13	10	11	0.062821		100/100		NonOverlappingTemplate
8	9	10	5	8	20	9	8	12	11	0.108791		96/100		NonOverlappingTemplate
12	4	8	5	14	12	8	16	11	10	0.162606		97/100		NonOverlappingTemplate
13	4	7	9	16	11	9	13	11	7	0.262249		98/100		NonOverlappingTemplate
10	7	8	4	9	11	13	11	7	20	0.048716		98/100		NonOverlappingTemplate
7	8	7	9	10	12	9	11	9	18	0.401199		98/100		NonOverlappingTemplate
11	6	8	10	4	16	13	12	11	9	0.289667		96/100		NonOverlappingTemplate
9	14	8	9	12	10	11	6	3	18	0.075719		100/100		NonOverlappingTemplate
9	7	12	4	13	17	9	7	10	12	0.202268		100/100		NonOverlappingTemplate
18	4	7	11	12	10	12	11	10	5	0.108791		98/100		NonOverlappingTemplate
6	12	9	9	9	15	7	12	8	13	0.595549		97/100		NonOverlappingTemplate
9	8	8	5	8	12	10	13	10	17	0.350485		98/100		NonOverlappingTemplate
8	4	7	5	11	14	9	22	5	15	0.000757		97/100		NonOverlappingTemplate
15	4	9	8	7	11	12	10	8	16	0.213309		96/100		NonOverlappingTemplate
17	14	9	5	9	5	10	12	7	12	0.145326		97/100		NonOverlappingTemplate
16	12	7	4	9	18	7	9	6	12	0.035174		96/100		NonOverlappingTemplate
5	6	13	7	11	13	4	12	10	19	0.025193		100/100		NonOverlappingTemplate
11	13	8	9	7	11	11	18	8	4	0.162606		98/100		NonOverlappingTemplate
4	10	8	6	12	9	15	10	13	13	0.319084		100/100		NonOverlappingTemplate
6	16	9	8	12	8	11	10	12	8	0.595549		98/100		NonOverlappingTemplate
11	12	12	8	10	9	10	11	9	8	0.991468		99/100		NonOverlappingTemplate
7	10	7	7	11	11	13	13	5	16	0.289667		100/100		NonOverlappingTemplate
7	5	9	5	12	9	12	15	8	18	0.062821		97/100		NonOverlappingTemplate
9	8	6	9	11	19	8	10	4	16	0.035174		99/100		NonOverlappingTemplate
5	11	8	7	10	10	9	16	11	13	0.474986		98/100		NonOverlappingTemplate
13	6	8	14	12	16	8	9	6	8	0.275709		97/100		NonOverlappingTemplate

7	8	9	7	15	12	15	9	8	10	0.514124	99/100	NonOverlappingTemplate
13	4	12	6	16	9	8	10	11	11	0.289667	97/100	NonOverlappingTemplate
9	4	9	8	10	12	11	7	11	19	0.129620	99/100	NonOverlappingTemplate
12	9	12	4	12	16	11	9	5	10	0.262249	98/100	NonOverlappingTemplate
7	9	5	12	7	12	10	14	12	12	0.574903	99/100	NonOverlappingTemplate
11	7	5	5	13	13	13	14	6	13	0.171867	99/100	NonOverlappingTemplate
12	10	4	5	12	10	15	9	7	16	0.122325	97/100	NonOverlappingTemplate
12	6	8	5	8	8	11	17	10	15	0.153763	99/100	NonOverlappingTemplate
5	5	9	7	9	17	13	12	10	13	0.153763	97/100	NonOverlappingTemplate
7	9	6	7	15	15	12	9	10	10	0.437274	100/100	NonOverlappingTemplate
15	7	3	9	15	6	7	9	10	19	0.010237	96/100	NonOverlappingTemplate
6	6	8	7	11	9	22	9	6	16	0.003712	100/100	NonOverlappingTemplate
11	9	6	6	7	11	11	11	13	15	0.534146	99/100	NonOverlappingTemplate
6	4	10	5	11	14	15	13	6	16	0.035174	98/100	NonOverlappingTemplate
5	8	9	7	7	13	11	10	14	16	0.275709	100/100	NonOverlappingTemplate
12	8	3	5	6	16	14	10	13	13	0.051942	98/100	NonOverlappingTemplate
10	12	14	3	3	12	10	8	10	18	0.025193	100/100	NonOverlappingTemplate
5	6	10	6	8	9	13	16	12	15	0.137282	100/100	NonOverlappingTemplate
10	8	8	7	13	13	8	9	7	17	0.366918	97/100	NonOverlappingTemplate
10	11	13	7	5	13	14	10	5	12	0.366918	98/100	NonOverlappingTemplate
10	9	6	8	14	7	16	7	8	15	0.213309	99/100	NonOverlappingTemplate
11	9	6	10	8	14	10	15	5	12	0.419021	98/100	NonOverlappingTemplate
7	9	10	2	16	8	17	10	9	12	0.051942	99/100	NonOverlappingTemplate
9	10	13	7	11	9	14	8	7	12	0.798139	96/100	NonOverlappingTemplate
12	7	9	11	8	14	8	10	10	11	0.911413	98/100	NonOverlappingTemplate
18	4	8	10	7	12	12	8	8	13	0.129620	97/100	NonOverlappingTemplate
7	6	9	8	14	8	15	13	8	12	0.419021	98/100	NonOverlappingTemplate
7	9	5	5	16	15	12	13	5	13	0.051942	100/100	NonOverlappingTemplate
6	4	6	9	11	9	9	13	15	18	0.048716	97/100	NonOverlappingTemplate
10	6	15	7	7	12	14	11	7	11	0.437274	97/100	NonOverlappingTemplate
5	6	9	6	13	12	12	7	10	20	0.030806	100/100	NonOverlappingTemplate
6	4	12	9	11	15	14	8	9	12	0.289667	98/100	NonOverlappingTemplate
19	6	6	5	14	9	14	8	11	8	0.035174	96/100	NonOverlappingTemplate
8	6	5	5	16	11	18	9	6	16	0.007694	97/100	NonOverlappingTemplate
6	12	10	8	12	8	11	8	6	19	0.145326	99/100	NonOverlappingTemplate
20	15	5	8	7	11	11	5	5	13	0.007694	92/100	* NonOverlappingTemplate
10	8	11	8	6	14	11	15	5	12	0.383827	99/100	NonOverlappingTemplate
10	11	8	2	10	12	18	10	5	14	0.037566	97/100	NonOverlappingTemplate
12	8	6	5	14	15	12	11	9	8	0.350485	99/100	NonOverlappingTemplate
10	5	11	7	15	12	11	8	5	16	0.162606	97/100	NonOverlappingTemplate
7	9	10	9	9	7	16	19	4	10	0.042808	98/100	NonOverlappingTemplate
12	6	3	6	8	9	11	21	11	13	0.008266	98/100	NonOverlappingTemplate
11	8	4	8	13	13	17	7	8	11	0.181557	99/100	NonOverlappingTemplate
7	11	4	4	13	14	12	14	8	13	0.122325	98/100	NonOverlappingTemplate
12	7	10	11	10	13	11	4	9	13	0.637119	98/100	NonOverlappingTemplate
12	3	10	3	7	14	18	13	8	12	0.013569	97/100	NonOverlappingTemplate
5	6	6	7	18	11	11	14	8	14	0.051942	100/100	NonOverlappingTemplate
10	3	9	9	14	11	14	9	10	11	0.474986	98/100	NonOverlappingTemplate
11	7	5	8	13	9	8	14	9	16	0.304126	99/100	NonOverlappingTemplate
10	6	12	7	8	11	15	16	9	6	0.262249	97/100	NonOverlappingTemplate
4	4	9	9	18	15	15	8	11	7	0.016717	100/100	NonOverlappingTemplate
14	9	15	3	11	10	13	9	5	11	0.171867	94/100	* NonOverlappingTemplate
6	7	5	12	7	13	13	9	12	16	0.202268	98/100	NonOverlappingTemplate
5	15	8	4	7	13	11	11	9	17	0.066882	99/100	NonOverlappingTemplate
1	8	6	9	15	14	14	9	10	14	0.040108	100/100	NonOverlappingTemplate
10	8	10	8	9	11	13	11	7	13	0.924076	99/100	NonOverlappingTemplate
7	9	6	6	8	6	9	15	14	20	0.015598	98/100	NonOverlappingTemplate
12	6	12	8	8	22	5	17	5	5	0.000439	99/100	NonOverlappingTemplate
16	9	13	10	8	10	6	9	6	13	0.419021	97/100	NonOverlappingTemplate
6	11	10	10	9	13	14	12	6	9	0.699313	99/100	NonOverlappingTemplate

13	7	6	5	6	13	6	16	12	16	0.040108	95/100 *	NonOverlappingTemplate
6	5	10	16	6	6	12	13	12	14	0.115387	98/100	NonOverlappingTemplate
5	14	11	9	8	8	15	12	9	9	0.514124	98/100	NonOverlappingTemplate
14	7	13	2	10	7	8	17	9	13	0.048716	97/100	NonOverlappingTemplate
8	6	12	4	10	19	9	8	12	12	0.080519	97/100	NonOverlappingTemplate
9	5	5	5	7	14	19	12	10	14	0.016717	99/100	NonOverlappingTemplate
9	11	10	7	11	9	14	6	13	10	0.798139	97/100	NonOverlappingTemplate
16	8	6	6	10	14	11	8	6	15	0.145326	99/100	NonOverlappingTemplate
7	11	11	6	9	12	12	9	8	15	0.678686	99/100	NonOverlappingTemplate
5	7	8	4	8	18	17	8	9	16	0.005762	100/100	NonOverlappingTemplate
5	7	9	2	10	16	9	14	12	16	0.023545	99/100	NonOverlappingTemplate
6	7	7	12	13	10	7	8	17	13	0.224821	99/100	NonOverlappingTemplate
4	10	8	5	17	10	11	8	11	16	0.075719	99/100	NonOverlappingTemplate
11	9	14	10	8	11	11	9	12	5	0.798139	99/100	NonOverlappingTemplate
8	6	9	7	12	10	11	10	12	15	0.699313	97/100	NonOverlappingTemplate
11	9	8	10	12	10	14	9	7	10	0.935716	98/100	NonOverlappingTemplate
10	8	7	7	11	8	6	14	14	15	0.350485	98/100	NonOverlappingTemplate
11	6	11	9	13	11	16	10	7	6	0.437274	93/100 *	NonOverlappingTemplate
4	7	13	10	18	11	13	8	3	13	0.025193	99/100	NonOverlappingTemplate
14	5	11	8	7	10	9	16	11	9	0.401199	99/100	NonOverlappingTemplate
9	6	7	12	6	13	11	13	12	11	0.637119	98/100	NonOverlappingTemplate
4	3	6	4	12	17	10	17	10	17	0.000700	100/100	NonOverlappingTemplate
10	12	7	8	8	7	11	14	8	15	0.574903	99/100	NonOverlappingTemplate
11	4	5	8	12	13	9	14	10	14	0.262249	96/100	NonOverlappingTemplate
14	8	10	4	10	10	14	9	10	11	0.595549	97/100	NonOverlappingTemplate
9	8	7	8	12	14	13	7	5	17	0.162606	98/100	NonOverlappingTemplate
14	12	5	11	8	6	9	15	9	11	0.401199	95/100 *	NonOverlappingTemplate
6	12	8	4	20	13	15	4	7	11	0.004301	98/100	NonOverlappingTemplate
16	10	8	6	12	10	8	9	10	11	0.678686	97/100	NonOverlappingTemplate
10	2	7	13	8	10	12	12	5	21	0.004301	97/100	NonOverlappingTemplate
14	7	9	6	9	13	4	8	12	18	0.066882	98/100	NonOverlappingTemplate
11	9	10	10	13	9	10	10	7	11	0.987896	96/100	NonOverlappingTemplate
8	13	10	4	11	7	21	6	13	7	0.010988	99/100	NonOverlappingTemplate
7	6	14	9	3	8	14	16	10	13	0.075719	98/100	NonOverlappingTemplate
11	9	11	8	10	8	12	12	5	14	0.739918	99/100	NonOverlappingTemplate
11	12	10	5	4	13	11	13	9	12	0.437274	99/100	NonOverlappingTemplate
12	8	8	6	11	5	14	14	7	15	0.213309	99/100	NonOverlappingTemplate
8	11	6	1	14	11	13	13	9	14	0.080519	98/100	NonOverlappingTemplate
9	8	4	9	15	13	9	18	6	9	0.071177	98/100	NonOverlappingTemplate
15	11	8	10	9	6	9	11	8	13	0.719747	98/100	NonOverlappingTemplate
11	11	8	9	6	14	9	14	10	8	0.739918	97/100	NonOverlappingTemplate
7	7	7	7	13	10	15	15	6	13	0.213309	97/100	NonOverlappingTemplate
6	4	6	8	11	10	7	13	16	19	0.013569	97/100	NonOverlappingTemplate
10	5	5	11	23	8	10	9	9	10	0.007160	99/100	OverlappingTemplate
100	0	0	0	0	0	0	0	0	0	0.000000 *	100/100	Universal
44	5	1	6	1	7	6	6	7	17	0.000000 *	71/100 *	ApproximateEntropy
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursions
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursions
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursions
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursions
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursions
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursions
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursions
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursions
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursionsVariant
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursionsVariant
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursionsVariant
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursionsVariant
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursionsVariant
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursionsVariant

```
  0   0   0   0   0   0   0   0   0   0    ----      ------     RandomExcursionsVariant
  0   0   0   0   0   0   0   0   0   0    ----      ------     RandomExcursionsVariant
  0   0   0   0   0   0   0   0   0   0    ----      ------     RandomExcursionsVariant
  0   0   0   0   0   0   0   0   0   0    ----      ------     RandomExcursionsVariant
  0   0   0   0   0   0   0   0   0   0    ----      ------     RandomExcursionsVariant
  0   0   0   0   0   0   0   0   0   0    ----      ------     RandomExcursionsVariant
  0   0   0   0   0   0   0   0   0   0    ----      ------     RandomExcursionsVariant
  0   0   0   0   0   0   0   0   0   0    ----      ------     RandomExcursionsVariant
  0   0   0   0   0   0   0   0   0   0    ----      ------     RandomExcursionsVariant
  0   0   0   0   0   0   0   0   0   0    ----      ------     RandomExcursionsVariant
  0   0   0   0   0   0   0   0   0   0    ----      ------     RandomExcursionsVariant
  0   0   0   0   0   0   0   0   0   0    ----      ------     RandomExcursionsVariant
  6   5   1   4   2   5   6  12  11  48  0.000000 *   97/100      Serial
 19   5   4   4   5   4   8   0   7  44  0.000000 *   91/100   *  Serial
100   0   0   0   0   0   0   0   0   0  0.000000 *    0/100   *  LinearComplexity


- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
The minimum pass rate for each statistical test with the exception of the
random excursion (variant) test is approximately = 96 for a
sample size = 100 binary sequences.

The minimum pass rate for the random excursion (variant) test is undefined.

For further guidelines construct a probability table using the MAPLE program
provided in the addendum section of the documentation.
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

Επεξήγηση τρόπου παρουσίασης συνοπτικών αποτελεσμάτων από τη σουίτα:

Όπως βλέπουμε τα αποτελέσματα εμφανίζονται, ουσιαστικά, με μια μορφή πίνακα μ γραμμών και ν=13 στηλών.

Ο αριθμός μ μας δίνει το πλήθος των στατιστικών ελέγχων που διενεργήθηκαν. Ενώ για τις ν στήλες έχουμε:

- οι στήλες 1-10 αντιπροσωπεύουν την συχνότητα των P-values σε καθένα από τα 10 ίσα υποδιαστήματα στα οποία έχει χωριστεί το διάστημα [0,1], αυτό γίνεται για να μπορέσει να ελεγχθεί αν η διαταραχή των P-values είναι ¨κοντά¨ στην ομοιόμορφη στο [0,1].
- η στήλη 11 είναι η στήλη που προκύπτει από την εφαρμογή ενός X^2 τεστ, για να μπορέσει να εκτιμηθεί η ομοιομορφία των P-values στον μ-οστό έλεγχο. Η τιμή αυτή είναι απροσδιόριστη αν το πλήθος των ακολουθιών που ελέγχεται είναι μικρότερο του 10. Εδώ πρέπει να τονιστεί ότι πρέπει να ελεγχθούν τουλάχιστον 55 ακολουθίες μιας γεννήτριας για να έχει στατιστικώς νόημα να ελέγξουμε την ομοιομορφία των P-values.
- η στήλη 12 είναι ιδιαίτερα σημαντική γιατί μας δίνει την αναλογία των ακολουθιών που πέρασαν τον μ-οστό έλεγχο.
- η στήλη 13 μας δείχνει ποιος είναι ο στατιστικός έλεγχος που λαμβάνει χώρα (όπως αυτοί έχουν οριστεί στο Κεφάλαιο 2).

Η ελάχιστη αναλογία που πρέπει να επιτυγχάνεται για να θεωρείται ότι μια ακολουθία πέρασε επιτυχώς τον εκάστοτε έλεγχο (εκτός του Random Excursions Variant) είναι (περίπου) 96/100 (στήλη 12 της συνοπτικής παρουσίασης). Εδώ πρέπει να τονιστεί ότι ζητώντας η τιμή της υπολογιζόμενης σε έναν έλεγχο P-value να είναι μεγαλύτερη ή ίση του 0.01 για να θεωρήσουμε επιτυχία, σημαίνει ότι στο συγκεκριμένο έλεγχο αν εξετάζονταν 100 τυχαίες ακολουθίες μια από αυτές θα απορριπτόταν και ο έλεγχος θα έβγαζε αποτυχία

για αυτήν. Αν αντίστοιχα θέταμε το όριο επιτυχίας σε 0.001, θα σημαίνει το αντίστοιχο για 1 στις 1000 τυχαίες ακολουθίες. Στην παρούσα έχουμε θέσει το όριο επιτυχίας στο 0.01 (για να έχουν νόημα τα αποτελέσματα θα πρέπει να ανήκει στο [0.01,0.001].

Τα προαναφερθέντα θα ισχύουν και στη συνέχεια της παρούσης κάθε φορά που θα εμφανίζεται συνοπτική παρουσίαση παρουσίαση αποτελεσμάτων με την άνω μορφή.

Συμπεράσματα: Η γεννήτρια αποτυγχάνει πλήρως στα τεστ: Binary Matrix Rank και Linear Complexity. Εδώ πρέπει να τονιστεί ότι το Linear Complexity test είναι ειδικά σχεδιασμένο για LFSRs και έχει σκοπό να ελέγξει αν ο καταχωρητής είναι αρκετά σύνθετος έτσι ώστε να παράγει μια ακολουθία που να θεωρείται τυχαία. Η πλήρης αποτυχία δείχνει ότι ο καταχωρητής που χρησιμοποιήθηκε δεν είναι αρκετά πολύπλοκος. Αυτό οφείλεται στο γεγονός ότι παρά το μεγάλο μήκος του δεν χρησιμοποιεί πολλά στοιχεία ως ορίσματα στην συνάρτηση ανάδρασης h. Αντίστοιχο αποτέλεσμα θα περιμέναμε και αν χρησιμοποιούσαμε και κάποιον καταχωρητή μικρού μήκους. Η πλήρης αποτυχία στο Binary Matrix Rank test σημαίνει ότι υπάρχει τεράστια γραμμική εξάρτηση ανάμεσα στις υπακολουθίες των ακολουθιών που παράγει κάθε φορά. Ο DFT παρότι είχε αποτύχει πλήρως στο κύριο παράδειγμα, επέτυχε αρκετά καλύτερη αναλογία στο δείγμα των 100 ελέγχων, αλλά και πάλι είναι κατά πολύ εκτός των επιθυμητών ορίων, αυτό μας δείχνει ότι η γεννήτρια παρουσιάζει σημαντικά περιοδικά χαρακτηριστικά. Η μη εκτέλεση των Random Excursions και Random Excursions Variant ήταν αναμενόμενη λόγω του μήκους των ακολουθιών που επιλέχθηκαν. Σημαντική πτώση παρουσιάζεται στις πολλαπλές ακολουθίες στο Approximate Entropy - αλλά είναι αναμενόμενη γιατί λόγω δεδομένων το τεστ δεν εκτελείται σωστά-και στη P-value2 του Serial, αλλά λόγω του πολύ καλού (τέλειου) αποτελέσματος στο κύριο παράδειγμα συστήνεται ο επανέλεγχος για μεγάλο δείγμα ακολουθιών (συνιστάται 1000) με μήκος της τάξεως του 10^6 και παράλληλη αλλαγή των αποδεκτών P-values από ≥ 0.01 σε ≥ 0.001.

4.2 Αξιολόγηση ΓΨΑ παραλλαγής πρωταρχικού LFSR μεγέθους 20-bit

Μια σημαντική κριτική που ασκείται σε έναν LFSR που δίνει ως έξοδο το λιγότερο σημαντικό bit κάθε κατάστασης είναι ότι στις πρώτες εξόδους του, δίνει ένα προς ένα τα στοιχεία του σπόρου, μέχρι αυτά να εξαντληθούν και να αρχίσει να δίνει εξόδους στοιχεία που έχουν προκύψει από τη συνάρτηση ανάδρασης h. Για αυτό τον λόγο θα εξεταστεί μια παραλλαγή του LFSR που εξετάστηκε προηγουμένως, η οποία έχει ως μόνη διαφορά ότι δίνει ως έξοδο σε κάθε παλμό του ρολογιού το αποτέλεσμα της συνάρτησης h Το χαρακτηριστικό πολυώνυμο, η αρχική κατάσταση και οι μετατοπίσεις είναι ακριβώς τα ίδια με προηγουμένως. Δημιουργήθηκε έτσι μια ακολουθία 1048575 bits, η οποία αποτελείται από 524288 το πλήθος άσσους και 524287 το πλήθος μηδενικά.

Τεστ	P-value
Frequency (Monobit)	0.999221 (ΕΠΙΤΥΧΙΑ)
Frequency within a block ($M = 128$)	0.692776 (ΕΠΙΤΥΧΙΑ)
Runs	0.997662 (ΕΠΙΤΥΧΙΑ)
Longest Run of Ones in a block ($M = 10^4, N = 104$)	0.945406 (ΕΠΙΤΥΧΙΑ)
Binary Matrix Rank	0.000000 (ΑΠΟΤΥΧΙΑ)
DFT (Spectral)	0.000000 (ΑΠΟΤΥΧΙΑ)
Non-Overlapping Template Matching	Για m=9 υπολογίζονται 148 P-values δείτε αναλυτικά παρακάτω
Overlapping Template Matching ($m = 9$)	0.046318 (ΕΠΙΤΥΧΙΑ)
Maurer's "Universal Statistical" $(L = 7, Q = 1280, K = 148516)$	0.444489 (ΕΠΙΤΥΧΙΑ)
Linear Complexity (M=500)	0.000000 (ΑΠΟΤΥΧΙΑ)
Serial ($m = 16$) P-value1	1.000000 (ΕΠΙΤΥΧΙΑ)
Serial ($m = 16$) P-value2	1.000000 (ΕΠΙΤΥΧΙΑ)
Approximate Entropy ($m = 10$)	1.000000 (ΕΠΙΤΥΧΙΑ)
Cumulative Sums (Cusum) forward	0.606164 (ΕΠΙΤΥΧΙΑ)
Cumulative Sums (Cusum) reverse	0.607045 (ΕΠΙΤΥΧΙΑ)
Random Excursions	Υπολογίζονται 8 P-values δείτε αναλυτικά παρακάτω
Random Excursions Variant	Υπολογίζονται 18 P-values δείτε αναλυτικά παρακάτω

Για το Non-overlapping Template Matching έχουμε για τα μη περιοδικά μοτίβο που εξετάστηκαν:

Μοτίβο (B)	P-value	Αποτέλεσμα	Α/α
000000001	0.639532	ΕΠΙΤΥΧΙΑ	0
000000011	0.720534	ΕΠΙΤΥΧΙΑ	1
000000101	0.522095	ΕΠΙΤΥΧΙΑ	2
000000111	0.215449	ΕΠΙΤΥΧΙΑ	3
000001001	0.025563	ΕΠΙΤΥΧΙΑ	4
000001011	0.325370	ΕΠΙΤΥΧΙΑ	5
000001101	0.747812	ΕΠΙΤΥΧΙΑ	6
000001111	0.220321	ΕΠΙΤΥΧΙΑ	7
000010001	0.511780	ΕΠΙΤΥΧΙΑ	8
000010011	0.792946	ΕΠΙΤΥΧΙΑ	9
000010101	0.930886	ΕΠΙΤΥΧΙΑ	10
000010111	0.562329	ΕΠΙΤΥΧΙΑ	11
000011001	0.531621	ΕΠΙΤΥΧΙΑ	12
000011011	0.993684	ΕΠΙΤΥΧΙΑ	13
000011101	0.647678	ΕΠΙΤΥΧΙΑ	14
000011111	0.684732	ΕΠΙΤΥΧΙΑ	15
000100011	0.631389	ΕΠΙΤΥΧΙΑ	16
000100101	0.381705	ΕΠΙΤΥΧΙΑ	17
000100111	0.700907	ΕΠΙΤΥΧΙΑ	18
000101001	0.628675	ΕΠΙΤΥΧΙΑ	19
000101011	0.872793	ΕΠΙΤΥΧΙΑ	20
000101101	0.765958	ΕΠΙΤΥΧΙΑ	21
000101111	0.042610	ΕΠΙΤΥΧΙΑ	22
000110011	0.077546	ΕΠΙΤΥΧΙΑ	23
000110101	0.921740	ΕΠΙΤΥΧΙΑ	24
000110111	0.890964	ΕΠΙΤΥΧΙΑ	25
000111001	0.413982	ΕΠΙΤΥΧΙΑ	26
000111011	0.928531	ΕΠΙΤΥΧΙΑ	27
000111101	0.631389	ΕΠΙΤΥΧΙΑ	28
000111111	0.279781	ΕΠΙΤΥΧΙΑ	29
001000011	0.184575	ΕΠΙΤΥΧΙΑ	30
001000101	0.613322	ΕΠΙΤΥΧΙΑ	31

001000111 0.483030 ΕΠΙΤΥΧΙΑ 32
001001011 0.124455 ΕΠΙΤΥΧΙΑ 33
001001101 0.711635 ΕΠΙΤΥΧΙΑ 34
001001111 0.641342 ΕΠΙΤΥΧΙΑ 35
001010011 0.197270 ΕΠΙΤΥΧΙΑ 36
001010101 0.140205 ΕΠΙΤΥΧΙΑ 37
001010111 0.539462 ΕΠΙΤΥΧΙΑ 38
001011011 0.759528 ΕΠΙΤΥΧΙΑ 39
001011101 0.837683 ΕΠΙΤΥΧΙΑ 40
001011111 0.499006 ΕΠΙΤΥΧΙΑ 41
001100101 0.153741 ΕΠΙΤΥΧΙΑ 42
001100111 0.829186 ΕΠΙΤΥΧΙΑ 43
001101011 0.384640 ΕΠΙΤΥΧΙΑ 44
001101101 0.826842 ΕΠΙΤΥΧΙΑ 45
001101111 0.984175 ΕΠΙΤΥΧΙΑ 46
001110101 0.945127 ΕΠΙΤΥΧΙΑ 47
001110111 0.251399 ΕΠΙΤΥΧΙΑ 48
001111011 0.583681 ΕΠΙΤΥΧΙΑ 49
001111101 0.675714 ΕΠΙΤΥΧΙΑ 50
001111111 0.625059 ΕΠΙΤΥΧΙΑ 51
010000011 0.799548 ΕΠΙΤΥΧΙΑ 52
010000111 0.258558 ΕΠΙΤΥΧΙΑ 53
010001011 0.422491 ΕΠΙΤΥΧΙΑ 54
010001111 0.708065 ΕΠΙΤΥΧΙΑ 55
010010011 0.532490 ΕΠΙΤΥΧΙΑ 56
010010111 0.592634 ΕΠΙΤΥΧΙΑ 57
010011011 0.822912 ΕΠΙΤΥΧΙΑ 58
010011111 0.308566 ΕΠΙΤΥΧΙΑ 59
010100011 0.887692 ΕΠΙΤΥΧΙΑ 60
010100111 0.379513 ΕΠΙΤΥΧΙΑ 61
010101011 0.944442 ΕΠΙΤΥΧΙΑ 62
010101111 0.313670 ΕΠΙΤΥΧΙΑ 63
010110011 0.325370 ΕΠΙΤΥΧΙΑ 64
010110111 0.213525 ΕΠΙΤΥΧΙΑ 65
010111011 0.276254 ΕΠΙΤΥΧΙΑ 66
010111111 0.203647 ΕΠΙΤΥΧΙΑ 67
011000111 0.881019 ΕΠΙΤΥΧΙΑ 68

011001111 0.717869 ΕΠΙΤΥΧΙΑ 69
011010111 0.393529 ΕΠΙΤΥΧΙΑ 70
011011111 0.889660 ΕΠΙΤΥΧΙΑ 71
011101111 0.938958 ΕΠΙΤΥΧΙΑ 72
011111111 0.182878 ΕΠΙΤΥΧΙΑ 73
100000000 0.639532 ΕΠΙΤΥΧΙΑ 74
100010000 0.739070 ΕΠΙΤΥΧΙΑ 75
100100000 0.255235 ΕΠΙΤΥΧΙΑ 76
100101000 0.822912 ΕΠΙΤΥΧΙΑ 77
100110000 0.892260 ΕΠΙΤΥΧΙΑ 78
100111000 0.488483 ΕΠΙΤΥΧΙΑ 79
101000000 0.747812 ΕΠΙΤΥΧΙΑ 80
101000100 0.957605 ΕΠΙΤΥΧΙΑ 81
101001000 0.280965 ΕΠΙΤΥΧΙΑ 82
101001100 0.615126 ΕΠΙΤΥΧΙΑ 83
101010000 0.282746 ΕΠΙΤΥΧΙΑ 84
101010100 0.637722 ΕΠΙΤΥΧΙΑ 85
101011000 0.653109 ΕΠΙΤΥΧΙΑ 86
101011100 0.579678 ΕΠΙΤΥΧΙΑ 87
101100000 0.732035 ΕΠΙΤΥΧΙΑ 88
101100100 0.977757 ΕΠΙΤΥΧΙΑ 89
101101000 0.633198 ΕΠΙΤΥΧΙΑ 90
101101100 0.902079 ΕΠΙΤΥΧΙΑ 91
101110000 0.443792 ΕΠΙΤΥΧΙΑ 92
101110100 0.566760 ΕΠΙΤΥΧΙΑ 93
101111000 0.906657 ΕΠΙΤΥΧΙΑ 94
101111100 0.780405 ΕΠΙΤΥΧΙΑ 95
110000000 0.766814 ΕΠΙΤΥΧΙΑ 96
110000010 0.499853 ΕΠΙΤΥΧΙΑ 97
110000100 0.516927 ΕΠΙΤΥΧΙΑ 98
110001000 0.342119 ΕΠΙΤΥΧΙΑ 99
110001010 0.280965 ΕΠΙΤΥΧΙΑ 100
110010000 0.661254 ΕΠΙΤΥΧΙΑ 101
110010010 0.447412 ΕΠΙΤΥΧΙΑ 102
110010100 0.849012 ΕΠΙΤΥΧΙΑ 103
110011000 0.080196 ΕΠΙΤΥΧΙΑ 104
110011010 0.340758 ΕΠΙΤΥΧΙΑ 105

110100000 0.889660 ΕΠΙΤΥΧΙΑ 106
110100010 0.699114 ΕΠΙΤΥΧΙΑ 107
110100100 0.269874 ΕΠΙΤΥΧΙΑ 108
110101000 0.389069 ΕΠΙΤΥΧΙΑ 109
110101010 0.516068 ΕΠΙΤΥΧΙΑ 110
110101100 0.858575 ΕΠΙΤΥΧΙΑ 111
110110000 0.779562 ΕΠΙΤΥΧΙΑ 112
110110010 0.899891 ΕΠΙΤΥΧΙΑ 113
110110100 0.485537 ΕΠΙΤΥΧΙΑ 114
110111000 0.664871 ΕΠΙΤΥΧΙΑ 115
110111010 0.741699 ΕΠΙΤΥΧΙΑ 116
110111100 0.679324 ΕΠΙΤΥΧΙΑ 117
111000000 0.147984 ΕΠΙΤΥΧΙΑ 118
111000010 0.372985 ΕΠΙΤΥΧΙΑ 119
111000100 0.860398 ΕΠΙΤΥΧΙΑ 120
111000110 0.642247 ΕΠΙΤΥΧΙΑ 121
111001000 0.865049 ΕΠΙΤΥΧΙΑ 122
111001010 0.887692 ΕΠΙΤΥΧΙΑ 123
111001100 0.085304 ΕΠΙΤΥΧΙΑ 124
111010000 0.419385 ΕΠΙΤΥΧΙΑ 125
111010010 0.781248 ΕΠΙΤΥΧΙΑ 126
111010100 0.385376 ΕΠΙΤΥΧΙΑ 127
111010110 0.244392 ΕΠΙΤΥΧΙΑ 128
111011000 0.700011 ΕΠΙΤΥΧΙΑ 129
111011010 0.477205 ΕΠΙΤΥΧΙΑ 130
111011100 0.923944 ΕΠΙΤΥΧΙΑ 131
111100000 0.396520 ΕΠΙΤΥΧΙΑ 132
111100010 0.998131 ΕΠΙΤΥΧΙΑ 133
111100100 0.787955 ΕΠΙΤΥΧΙΑ 134
111100110 0.641342 ΕΠΙΤΥΧΙΑ 135
111101000 0.207835 ΕΠΙΤΥΧΙΑ 136
111101010 0.267581 ΕΠΙΤΥΧΙΑ 137
111101100 0.593531 ΕΠΙΤΥΧΙΑ 138
111101110 0.672102 ΕΠΙΤΥΧΙΑ 139
111110000 0.081028 ΕΠΙΤΥΧΙΑ 140
111110010 0.947406 ΕΠΙΤΥΧΙΑ 141
111110100 0.468120 ΕΠΙΤΥΧΙΑ 142

111110110 0.932439 ΕΠΙΤΥΧΙΑ 143

111111000 0.111222 ΕΠΙΤΥΧΙΑ 144

111111010 0.535972 ΕΠΙΤΥΧΙΑ 145

111111100 0.212091 ΕΠΙΤΥΧΙΑ 146

111111110 0.182878 ΕΠΙΤΥΧΙΑ 147

Για το Random Excursions test (μετά από 1295 κύκλους J) έχουμε:

ΕΠΙΤΥΧΙΑ x = -4 P-value = 0.151393

ΕΠΙΤΥΧΙΑ x = -3 P-value = 0.085533

ΕΠΙΤΥΧΙΑ x = -2 P-value = 0.207332

ΕΠΙΤΥΧΙΑ x = -1 P-value = 0.223528

ΕΠΙΤΥΧΙΑ x = 1 P-value = 0.071998

ΕΠΙΤΥΧΙΑ x = 2 P-value = 0.139737

ΕΠΙΤΥΧΙΑ x = 3 P-value = 0.015666

ΕΠΙΤΥΧΙΑ x = 4 P-value = 0.039739

Για το Random Excursions Variant test (μετά από 1295 κύκλους J) έχουμε:

ΕΠΙΤΥΧΙΑ (x = -9) P-value = 0.745887

ΕΠΙΤΥΧΙΑ (x = -8) P-value = 0.939338

ΕΠΙΤΥΧΙΑ (x = -7) P-value = 0.674754

ΕΠΙΤΥΧΙΑ (x = -6) P-value = 0.451801

ΕΠΙΤΥΧΙΑ (x = -5) P-value = 0.428053

ΕΠΙΤΥΧΙΑ (x = -4) P-value = 0.475863

ΕΠΙΤΥΧΙΑ (x = -3) P-value = 0.660390

ΕΠΙΤΥΧΙΑ (x = -2) P-value = 0.759373

ΕΠΙΤΥΧΙΑ (x = -1) P-value = 0.409215

ΕΠΙΤΥΧΙΑ (x = 1) P-value = 0.115961

ΕΠΙΤΥΧΙΑ (x = 2) P-value = 0.052388

ΕΠΙΤΥΧΙΑ (x = 3) P-value = 0.010289

ΕΠΙΤΥΧΙΑ (x = 4) P-value = 0.014851

ΕΠΙΤΥΧΙΑ (x = 5) P-value = 0.059248

ΕΠΙΤΥΧΙΑ (x = 6) P-value = 0.158529

ΕΠΙΤΥΧΙΑ (x = 7) P-value = 0.313355

ΕΠΙΤΥΧΙΑ (x = 8) P-value = 0.428675

ΕΠΙΤΥΧΙΑ (x = 9) P-value = 0.246839

Όπως βλέπουμε τα αποτελέσματα της αναλυτικής παρουσίασης για την ακολουθία που παράχθηκε σε πολύ μεγάλο ποσοστό ίδια με αυτά στην 4.1 και ελαφρώς βελτιωμένα. Η ακολουθία αποτυγχάνει πλήρως στα τεστ: Binary Matrix Rank, DFT και Linear Complexity. Αλλά αυτή τη φορά επιτυγχάνουν όλες οι P-values στα τεστ Random Excursions και Random Excursions Variant. Για την καλύτερη αξιολόγηση της γεννήτριας υποβλήθηκαν πάλι σε έλεγχο 100 ακολουθίες μήκους 10480 bits η κάθε μια, που δημιουργήθηκαν από αυτή. Τα συνοπτικά αποτελέσματα όπως τα εμφανίζει η σουίτα είναι:

RESULTS FOR THE UNIFORMITY OF P-VALUES AND THE PROPORTION OF PASSING SEQUENCES

generator is <LFSR_20_3_lst_row_final.txt>

C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	P-VALUE		PROPORTION		STATISTICAL TEST
7	10	5	5	20	9	13	14	11	6	0.016717		99/100		Frequency
20	6	8	2	6	3	5	7	9	34	0.000000	*	91/100	*	BlockFrequency
6	11	8	7	7	12	12	12	10	15	0.574903		99/100		CumulativeSums
7	5	9	6	9	9	10	11	19	15	0.066882		99/100		CumulativeSums
6	10	8	13	4	16	12	9	12	10	0.275709		98/100		Runs
13	11	5	17	12	11	8	12	5	6	0.129620		100/100		LongestRun
100	0	0	0	0	0	0	0	0	0	0.000000	*	0/100	*	Rank
32	7	14	4	9	10	5	8	3	8	0.000000	*	87/100	*	FFT
9	6	11	8	10	11	14	11	6	14	0.616305		98/100		NonOverlappingTemplate
12	13	10	5	12	13	13	10	4	8	0.350485		97/100		NonOverlappingTemplate
16	16	8	9	11	9	9	8	6	8	0.319084		95/100	*	NonOverlappingTemplate
27	11	7	9	8	7	11	6	6	8	0.000060	*	94/100	*	NonOverlappingTemplate
12	10	12	11	10	15	10	4	5	11	0.383827		95/100	*	NonOverlappingTemplate
12	2	8	9	11	14	10	8	13	13	0.262249		99/100		NonOverlappingTemplate
8	12	5	9	6	14	12	13	11	10	0.534146		98/100		NonOverlappingTemplate
11	12	4	10	12	16	15	3	6	11	0.045675		96/100		NonOverlappingTemplate
8	7	12	6	7	13	21	6	8	12	0.020548		96/100		NonOverlappingTemplate
8	7	11	5	11	11	12	11	12	12	0.798139		99/100		NonOverlappingTemplate
12	7	9	13	9	5	10	13	7	15	0.419021		98/100		NonOverlappingTemplate
14	8	11	5	8	9	13	11	12	9	0.678686		96/100		NonOverlappingTemplate
11	8	9	9	10	12	14	11	6	10	0.883171		97/100		NonOverlappingTemplate
13	7	11	9	7	11	18	5	5	14	0.066882		97/100		NonOverlappingTemplate
7	7	8	8	5	15	13	15	5	17	0.030806		100/100		NonOverlappingTemplate
12	7	8	3	19	12	8	12	8	11	0.058984		98/100		NonOverlappingTemplate
5	12	10	7	11	6	7	11	14	17	0.162606		100/100		NonOverlappingTemplate
13	8	5	8	15	12	12	7	7	13	0.334538		97/100		NonOverlappingTemplate
6	8	9	6	12	15	16	6	9	13	0.171867		98/100		NonOverlappingTemplate
10	3	7	4	17	12	13	13	9	12	0.048716		100/100		NonOverlappingTemplate
9	7	11	5	10	19	8	9	8	14	0.115387		98/100		NonOverlappingTemplate
9	6	9	7	10	13	10	14	12	10	0.779188		97/100		NonOverlappingTemplate
12	6	10	9	13	10	9	16	7	8	0.534146		97/100		NonOverlappingTemplate
11	6	9	6	9	10	11	11	11	16	0.595549		98/100		NonOverlappingTemplate
7	9	8	6	11	14	9	9	10	17	0.366918		98/100		NonOverlappingTemplate
11	7	8	9	3	19	11	13	10	9	0.075719		96/100		NonOverlappingTemplate
8	16	7	11	10	10	7	8	5	18	0.085587		100/100		NonOverlappingTemplate
8	8	9	2	18	16	10	8	7	14	0.016717		100/100		NonOverlappingTemplate
18	5	5	13	13	6	11	12	13	4	0.019188		98/100		NonOverlappingTemplate
8	9	11	7	11	13	8	15	8	10	0.759756		98/100		NonOverlappingTemplate
12	4	10	7	8	10	15	7	9	18	0.085587		98/100		NonOverlappingTemplate
7	5	5	8	13	9	16	17	4	16	0.006196		97/100		NonOverlappingTemplate
13	8	8	8	8	15	9	8	7	16	0.350485		96/100		NonOverlappingTemplate
17	17	6	5	8	7	10	10	8	12	0.066882		97/100		NonOverlappingTemplate
14	15	8	6	8	15	7	12	4	11	0.122325		95/100	*	NonOverlappingTemplate
4	8	14	5	10	16	4	8	10	21	0.001030		100/100		NonOverlappingTemplate
15	9	10	6	7	12	13	16	7	5	0.145326		99/100		NonOverlappingTemplate
4	11	7	6	14	10	12	7	8	21	0.010237		100/100		NonOverlappingTemplate
6	14	10	10	9	11	9	11	11	9	0.924076		98/100		NonOverlappingTemplate
12	10	16	5	11	9	9	12	6	10	0.455937		100/100		NonOverlappingTemplate
8	8	11	5	11	10	10	17	6	14	0.236810		100/100		NonOverlappingTemplate
7	6	7	7	8	13	10	15	8	19	0.055361		97/100		NonOverlappingTemplate
10	8	6	9	14	12	12	9	4	16	0.224821		98/100		NonOverlappingTemplate
6	12	3	8	11	14	8	14	8	16	0.090936		99/100		NonOverlappingTemplate
14	7	6	10	11	17	11	11	7	6	0.224821		98/100		NonOverlappingTemplate

8	7	5	11	10	15	16	11	6	11	0.224821	99/100	NonOverlappingTemplate
10	8	10	6	14	14	10	7	10	11	0.719747	97/100	NonOverlappingTemplate
8	4	11	8	11	9	12	8	8	21	0.035174	99/100	NonOverlappingTemplate
15	7	8	7	14	12	9	12	6	10	0.455937	98/100	NonOverlappingTemplate
6	9	4	11	9	12	11	12	13	13	0.514124	100/100	NonOverlappingTemplate
13	3	6	9	9	13	15	10	7	15	0.108791	99/100	NonOverlappingTemplate
9	15	6	4	9	6	17	6	12	16	0.017912	98/100	NonOverlappingTemplate
12	8	5	7	9	8	11	14	10	16	0.350485	99/100	NonOverlappingTemplate
5	4	8	10	5	21	10	12	11	14	0.005762	97/100	NonOverlappingTemplate
4	13	6	4	16	14	11	15	7	10	0.030806	99/100	NonOverlappingTemplate
16	7	2	13	12	5	9	8	10	18	0.010237	96/100	NonOverlappingTemplate
7	5	8	9	10	9	16	14	3	19	0.008266	100/100	NonOverlappingTemplate
11	10	7	6	4	13	13	12	11	13	0.401199	99/100	NonOverlappingTemplate
6	4	12	6	8	17	14	10	5	18	0.006196	98/100	NonOverlappingTemplate
6	8	10	6	6	13	12	10	14	15	0.304126	100/100	NonOverlappingTemplate
8	14	3	3	5	20	10	10	15	12	0.001296	98/100	NonOverlappingTemplate
6	15	16	2	7	9	7	12	10	16	0.017912	100/100	NonOverlappingTemplate
4	7	11	3	12	8	13	14	11	17	0.037566	100/100	NonOverlappingTemplate
8	9	5	9	14	15	6	11	9	14	0.304126	97/100	NonOverlappingTemplate
10	12	15	6	6	8	15	9	5	14	0.153763	98/100	NonOverlappingTemplate
10	7	9	8	13	7	13	12	8	13	0.759756	99/100	NonOverlappingTemplate
12	8	9	7	12	11	10	12	6	13	0.816537	98/100	NonOverlappingTemplate
7	8	11	4	12	9	18	9	9	13	0.162606	99/100	NonOverlappingTemplate
10	9	8	9	12	10	14	10	7	11	0.935716	96/100	NonOverlappingTemplate
12	5	13	8	11	13	8	7	12	11	0.637119	97/100	NonOverlappingTemplate
16	6	9	9	7	8	13	8	12	12	0.455937	98/100	NonOverlappingTemplate
7	7	8	8	13	8	16	10	11	12	0.534146	99/100	NonOverlappingTemplate
7	9	6	2	18	12	13	15	4	14	0.003712	100/100	NonOverlappingTemplate
6	5	8	9	10	8	6	15	11	22	0.004981	97/100	NonOverlappingTemplate
9	6	11	8	11	11	13	11	5	15	0.494392	98/100	NonOverlappingTemplate
6	5	15	6	12	17	9	10	10	10	0.137282	98/100	NonOverlappingTemplate
18	3	11	6	14	8	11	10	12	7	0.058984	96/100	NonOverlappingTemplate
7	8	4	6	15	10	19	8	5	18	0.001757	97/100	NonOverlappingTemplate
5	10	10	6	12	12	12	8	6	19	0.080519	99/100	NonOverlappingTemplate
22	10	8	11	6	11	9	8	4	11	0.013569	92/100	* NonOverlappingTemplate
11	8	9	7	9	16	11	13	5	11	0.455937	99/100	NonOverlappingTemplate
10	10	9	6	6	11	18	10	6	14	0.162606	97/100	NonOverlappingTemplate
10	10	6	6	12	18	11	12	9	6	0.202268	99/100	NonOverlappingTemplate
8	6	12	7	16	13	8	9	8	13	0.383827	95/100	* NonOverlappingTemplate
6	8	16	6	9	8	14	17	7	9	0.085587	98/100	NonOverlappingTemplate
12	5	4	6	8	14	9	18	15	9	0.023545	98/100	NonOverlappingTemplate
11	7	6	6	16	9	17	8	9	11	0.145326	99/100	NonOverlappingTemplate
7	11	6	5	11	11	15	8	9	17	0.153763	98/100	NonOverlappingTemplate
10	8	10	11	13	10	13	8	7	10	0.935716	98/100	NonOverlappingTemplate
12	3	10	4	10	13	16	11	6	15	0.040108	97/100	NonOverlappingTemplate
5	6	7	6	15	12	13	13	8	15	0.115387	100/100	NonOverlappingTemplate
9	5	9	9	10	15	10	10	12	11	0.759756	97/100	NonOverlappingTemplate
9	9	5	7	11	14	7	14	9	15	0.319084	99/100	NonOverlappingTemplate
11	6	9	7	10	15	12	17	8	5	0.145326	98/100	NonOverlappingTemplate
4	5	10	7	16	10	22	7	10	9	0.002043	100/100	NonOverlappingTemplate
13	7	16	8	12	7	10	13	5	9	0.304126	95/100	* NonOverlappingTemplate
7	6	6	13	5	14	17	6	12	14	0.040108	98/100	NonOverlappingTemplate
7	10	9	3	11	14	10	10	9	17	0.181557	99/100	NonOverlappingTemplate
3	6	8	9	12	13	15	11	8	15	0.129620	100/100	NonOverlappingTemplate
11	6	10	10	8	11	10	13	9	12	0.935716	100/100	NonOverlappingTemplate
7	8	7	5	8	8	10	18	9	20	0.008879	99/100	NonOverlappingTemplate
14	5	13	9	8	18	10	13	6	4	0.035174	99/100	NonOverlappingTemplate
12	14	11	10	10	10	5	6	8	14	0.514124	96/100	NonOverlappingTemplate
5	11	10	9	10	11	15	12	8	9	0.719747	100/100	NonOverlappingTemplate
13	8	6	4	5	16	9	10	12	17	0.035174	96/100	NonOverlappingTemplate

6	5	11	14	8	9	9	19	4	15	0.014550	98/100	NonOverlappingTemplate
6	14	9	5	8	13	15	12	5	13	0.145326	98/100	NonOverlappingTemplate
12	8	12	5	11	6	6	10	15	15	0.213309	97/100	NonOverlappingTemplate
10	6	11	4	6	19	10	12	12	10	0.071177	96/100	NonOverlappingTemplate
9	4	3	6	8	11	22	16	8	13	0.000439	98/100	NonOverlappingTemplate
7	10	10	8	11	6	14	15	10	9	0.616305	97/100	NonOverlappingTemplate
15	8	6	6	9	12	12	9	6	17	0.137282	99/100	NonOverlappingTemplate
7	12	10	5	8	15	17	2	7	17	0.004629	99/100	NonOverlappingTemplate
5	5	7	3	12	18	15	9	8	18	0.001399	100/100	NonOverlappingTemplate
5	9	8	3	7	15	12	14	9	18	0.019188	99/100	NonOverlappingTemplate
6	9	6	8	15	7	9	13	14	13	0.304126	99/100	NonOverlappingTemplate
4	9	10	7	16	8	9	10	8	19	0.045675	99/100	NonOverlappingTemplate
12	8	10	13	11	10	10	10	12	4	0.759756	99/100	NonOverlappingTemplate
8	5	11	7	10	12	10	12	9	16	0.494392	97/100	NonOverlappingTemplate
11	9	11	8	13	8	12	10	9	9	0.978072	98/100	NonOverlappingTemplate
8	7	4	10	12	9	10	11	14	15	0.383827	98/100	NonOverlappingTemplate
12	5	11	9	14	12	14	11	6	6	0.350485	93/100 *	NonOverlappingTemplate
4	7	15	13	11	15	8	10	4	13	0.080519	99/100	NonOverlappingTemplate
13	9	7	7	11	7	6	18	10	12	0.202268	99/100	NonOverlappingTemplate
9	7	6	8	13	10	9	13	12	13	0.719747	98/100	NonOverlappingTemplate
4	4	7	3	14	11	6	20	12	19	0.000065 *	100/100	NonOverlappingTemplate
11	11	6	10	7	8	11	15	6	15	0.366918	99/100	NonOverlappingTemplate
9	4	6	11	11	9	10	16	8	16	0.153763	96/100	NonOverlappingTemplate
13	10	8	5	11	12	13	7	9	12	0.678686	98/100	NonOverlappingTemplate
8	9	8	5	12	17	14	6	5	16	0.035174	100/100	NonOverlappingTemplate
15	9	4	10	7	10	10	16	10	9	0.289667	94/100 *	NonOverlappingTemplate
7	8	10	5	16	17	11	7	9	10	0.145326	98/100	NonOverlappingTemplate
16	9	11	7	12	8	5	9	12	11	0.474986	97/100	NonOverlappingTemplate
10	3	8	11	8	12	6	14	5	23	0.000700	97/100	NonOverlappingTemplate
14	8	7	9	7	14	3	6	14	18	0.017912	98/100	NonOverlappingTemplate
13	7	10	9	10	10	12	9	8	12	0.955835	96/100	NonOverlappingTemplate
9	10	12	5	8	6	22	11	9	8	0.017912	98/100	NonOverlappingTemplate
6	4	15	6	9	15	15	8	9	13	0.071177	98/100	NonOverlappingTemplate
10	9	9	7	11	8	15	13	4	14	0.334538	99/100	NonOverlappingTemplate
11	13	9	3	8	10	15	13	8	10	0.334538	99/100	NonOverlappingTemplate
11	6	10	6	11	7	12	12	10	15	0.574903	99/100	NonOverlappingTemplate
7	13	5	3	14	12	12	11	10	13	0.181557	98/100	NonOverlappingTemplate
9	8	7	7	17	8	14	14	7	9	0.224821	98/100	NonOverlappingTemplate
14	11	10	9	9	8	7	13	8	11	0.867692	98/100	NonOverlappingTemplate
11	11	9	11	5	13	11	13	8	8	0.779188	98/100	NonOverlappingTemplate
7	5	11	6	10	9	19	11	7	15	0.051942	97/100	NonOverlappingTemplate
6	5	8	9	10	8	6	15	10	23	0.002043	97/100	NonOverlappingTemplate
7	8	7	9	19	9	8	10	11	12	0.249284	100/100	OverlappingTemplate
100	0	0	0	0	0	0	0	0	0	0.000000 *	100/100	Universal
44	5	2	4	4	4	6	6	8	17	0.000000 *	72/100 *	ApproximateEntropy
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursions
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursions
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursions
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursions
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursions
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursions
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursions
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursions
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursionsVariant
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursionsVariant
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursionsVariant
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursionsVariant
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursionsVariant
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursionsVariant
0	0	0	0	0	0	0	0	0	0	----	------	RandomExcursionsVariant

0	0	0	0	0	0	0	0	0	0	----		------		RandomExcursionsVariant
0	0	0	0	0	0	0	0	0	0	----		------		RandomExcursionsVariant
0	0	0	0	0	0	0	0	0	0	----		------		RandomExcursionsVariant
0	0	0	0	0	0	0	0	0	0	----		------		RandomExcursionsVariant
0	0	0	0	0	0	0	0	0	0	----		------		RandomExcursionsVariant
0	0	0	0	0	0	0	0	0	0	----		------		RandomExcursionsVariant
0	0	0	0	0	0	0	0	0	0	----		------		RandomExcursionsVariant
0	0	0	0	0	0	0	0	0	0	----		------		RandomExcursionsVariant
0	0	0	0	0	0	0	0	0	0	----		------		RandomExcursionsVariant
0	0	0	0	0	0	0	0	0	0	----		------		RandomExcursionsVariant
0	0	0	0	0	0	0	0	0	0	----		------		RandomExcursionsVariant
7	3	2	4	1	6	7	9	14	47	0.000000	*	97/100		Serial
19	5	2	6	2	7	6	2	9	42	0.000000	*	91/100	*	Serial
100	0	0	0	0	0	0	0	0	0	0.000000	*	0/100	*	LinearComplexity

- -

The minimum pass rate for each statistical test with the exception of the random excursion (variant) test is approximately = 96 for a sample size = 100 binary sequences.

The minimum pass rate for the random excursion (variant) test is undefined.

For further guidelines construct a probability table using the MAPLE program provided in the addendum section of the documentation.

- -

<u>Συμπεράσματα</u>: Όπως ήταν αναμενόμενο μετά τον αναλυτικό έλεγχο της πρώτης ακολουθίας που παράχθηκε, βλέπουμε ότι η γεννήτρια έχει πανομοιότυπη συμπεριφορά ως προς τους ελέγχους με αυτή του 4.1. Πιο συγκεκριμένα αποτυγχάνει πλήρως στα τεστ: Binary Matrix Rank Linear Complexity. Η γεννήτρια αποτυγχάνει πλήρως στα τεστ: Binary Matrix Rank και Linear Complexity. Εδώ πρέπει να τονιστεί ότι το Linear Complexity test είναι ειδικά σχεδιασμένο για LFSRs και έχει σκοπό να ελέγξει αν ο καταχωρητής είναι αρκετά σύνθετος έτσι ώστε να παράγει μια ακολουθία που να θεωρείται τυχαία. Η πλήρης αποτυχία δείχνει ότι ο καταχωρητής που χρησιμοποιήθηκε δεν είναι αρκετά πολύπλοκος. Αυτό οφείλεται στο γεγονός ότι παρά το μεγάλο μήκος του δεν χρησιμοποιεί πολλά στοιχεία ως ορίσματα στην συνάρτηση ανάδρασης h. Αντίστοιχο αποτέλεσμα θα περιμέναμε και αν χρησιμοποιούσαμε και κάποιον καταχωρητή μικρού μήκους. Η πλήρης αποτυχία στο Binary Matrix Rank test σημαίνει ότι υπάρχει τεράστια γραμμική εξάρτηση ανάμεσα στις υπακολουθίες των ακολουθιών που παράγει κάθε φορά. Ο DFT παρότι είχε αποτύχει πλήρως στο κύριο παράδειγμα, επέτυχε αρκετά καλύτερη αναλογία στο δείγμα των 100 ελέγχων, αλλά και πάλι είναι κατά πολύ εκτός των επιθυμητών ορίων, αυτό μας δείχνει ότι η γεννήτρια παρουσιάζει σημαντικά περιοδικά χαρακτηριστικά. Η μη εκτέλεση των Random Excursions και Random Excursions Variant ήταν αναμενόμενη λόγω του μήκους των ακολουθιών που επιλέχθηκαν. Σημαντική πτώση παρουσιάζεται στις πολλαπλές ακολουθίες στο Approximate Entropy -αλλά είναι αναμενόμενη γιατί λόγω δεδομένων το τεστ δεν εκτελείται σωστά-και στη P-value2 του Serial, αλλά λόγω του πολύ καλού (τέλειου) αποτελέσματος στο κύριο παράδειγμα συστήνεται ο επανέλεγχος για μεγάλο δείγμα ακολουθιών (συνιστάται 1000) με μήκος της τάξεως του 10^6 και παράλληλη αλλαγή των αποδεκτών P-values από ≥ 0.01 σε ≥ 0.001.

4.3 Αξιολόγηση ΓΨΑ Geffe

Δημιουργήθηκε και εξετάστηκε παράδειγμα ακολουθίας ψευδοτυχαίων αριθμών που παράγεται από ΓΨΑ Geffe με συνάρτηση πολλαπλασιασμού

$F(\chi_1,\chi_2,\chi_3) = \chi_1\chi_2 \oplus \chi_2\chi_3 \oplus \chi_3$, στην οποία χρησιμοποιούνται ως είσοδο οι έξοδοι τριών LFSRs (βλέπε 3.8). Για να καταστεί δυνατόν στον αναγνώστη να δει σε ένα παράδειγμα την ακολουθία που εξετάζεται αποφασίστηκε η ακολουθία αυτή να είναι μεγέθους 2048 bits, έτσι ώστε να μπορεί να παρουσιαστεί στην παρούσα (δυστυχώς ακολουθίες της τάξης 10^6 bits απαιτούν περίπου 315 σελίδες για να αποτυπωθούν). Η ακολουθία που παράχθηκε, λοιπόν, είναι η:

ε=

011100010100000001101011111100110100001011100110010011001001110011010110011000
100111010110110000001110100110100001010111011000101110101111101111011101101100 1
000010100100110100010011111011100000100011101011111100010000111110101111000010 1
111001000111010001000101111110110010100010111000111101001001011100011110100001 10
001100100010000001010101001111000100000110000101111010100001100011001101010101 0
100111010111111011000011101100011000110100010100011001011111010100001011011100 1
011000000001100010000000101101011011001011001001100101100000111100001000001101 1
110011111111000110000010000110100101001011011111000000011111011111000000010110 1
000110010011100101010011001011101100100011100101001110101101001000001011101001 1
101011011000101011100000001101111011011100110100110101101110111111011010110000 10
011000111111100000000110001010111010001101010011011110100010000111100001101110 1
111111011011111001101110110111111111011000010101000100100010101110111100011010 10
011001010101110011111111110011001110100001111011001101010101110010001001101101 01
010100111001010100100011011011101001000010000010100110110000001010100111110100 1
010111010011110100001010001010101111011111111110001001100111101100000000101001 111
100111011110010101101101111011110001001100110101001100101000110101010011110010 0
000001011110111111000101001110111110010110010011010110001111001000111100110101 0
110001011111011100000110110011101010000111001111110000101010100001000100110010 0
110110000010100111010101110001110101001101010111111000010100111001001010100101 1
111001101101000111010110000011111111011001010111001010100110100110110011111010 1
111011001100101010100111111011111000010101010010110100110010110011010111111111 1
100000110110111111101111101001100111000001101001100101000100001001101011010000 1
101000001110100100011001101110000101110011111000011110100011001010011000111000 10
110011101101111011001001001101000011010100110001000000100101010101100101101100 0
110001101100001010101100100000101001100000001000010000100010111100110001100000 1
110010010001110011111111011010000000101111110000100110010000011111111111110.

η οποία αποτελείται από 1040 το πλήθος άσσους και 1008 το πλήθος μηδενικά.

Και τα αποτελέσματα των ελέγχων που διενεργήθηκαν σε αυτή αναφέρονται κάτωθι.

Τεστ	P-value
Frequency (Monobit)	0.479500 (ΕΠΙΤΥΧΙΑ)
Frequency within a block ($M = 128$)	0.677322 (ΕΠΙΤΥΧΙΑ)
Runs	0.748600 (ΕΠΙΤΥΧΙΑ)
Longest Run of Ones in a block ($N = 256, M = 8$)	0.125026 (ΕΠΙΤΥΧΙΑ)
Binary Matrix Rank	0.741908 (ΕΠΙΤΥΧΙΑ)
DFT (Spectral)	0.655519 (ΕΠΙΤΥΧΙΑ)
Non-Overlapping Template Matching	Για $m = 9$ υπολογίζονται 148 P-values δείτε αναλυτικά παρακάτω
Overlapping Template Matching ($m = 9$)	0.488416 (ΕΠΙΤΥΧΙΑ)
Maurer's "Universal Statistical"	ΣΦΑΛΜΑ
Linear Complexity (M=500)	0.423166 (ΕΠΙΤΥΧΙΑ)
Serial ($m = 16$) P-value1	0.597762 (ΕΠΙΤΥΧΙΑ)
Serial ($m = 16$) P-value2	0.533764 (ΕΠΙΤΥΧΙΑ)
Approximate Entropy (m=10)	0.010990 (ΕΠΙΤΥΧΙΑ)
Cumulative Sums (Cusum) forward	0.277459 (ΕΠΙΤΥΧΙΑ)
Cumulative Sums (Cusum) reverse	0.415360 (ΕΠΙΤΥΧΙΑ)
Random Excursions	ΜΗ ΕΦΑΡΜΟΣΙΜΟ
Random Excursions Variant	ΜΗ ΕΦΑΡΜΟΣΙΜΟ

Σχόλια:

1. Στο Maurer's "Universal Statistical" test εμφανίζεται σφάλμα, γιατί , όπως περιγράφεται και στο 2.9, απαιτεί το μήκος της υπό εξέταση ακολουθίας να είναι ≥ 387840 bits, για να ξεκινήσει (να δώσει τιμές στις μεταβλητές L, Q και να ξεκινήσει η διαδικασία). Επομένως, επειδή δίνεται ακολουθία μήκους 2048 (bits), δεν μπορεί να ξεκινήσει.
2. Τα τεστ Random Excursions και Random Excursions Variant δεν μπορούν να εφαρμοστούν γιατί δημιουργούνται μόνο 37 κύκλοι (J) με αποτέλεσμα να έχουμε 8 στο πρώτο και 18 στο δεύτερο (τεστ) P-values ίσες με 0.000000. Αναμενόμενο λόγο του μικρού μήκους της ακολουθίας (όπως αναφέρεται και στις 2.14. 2.15 συνιστάται μήκος$\geq 10^6$).

Για το Non-overlapping Template Matching έχουμε για τα μη περιοδικά μοτίβο που εξετάστηκαν:

Μοτίβο	P-value	Αποτέλεσμα	Α/α
000000001	0.856028	ΕΠΙΤΥΧΙΑ	0
000000011	0.400661	ΕΠΙΤΥΧΙΑ	1
000000101	0.128476	ΕΠΙΤΥΧΙΑ	2
000000111	0.856028	ΕΠΙΤΥΧΙΑ	3
000001001	0.861831	ΕΠΙΤΥΧΙΑ	4
000001011	0.400661	ΕΠΙΤΥΧΙΑ	5
000001101	0.128476	ΕΠΙΤΥΧΙΑ	6
000001111	0.406722	ΕΠΙΤΥΧΙΑ	7
000010001	0.850132	ΕΠΙΤΥΧΙΑ	8
000010011	0.850132	ΕΠΙΤΥΧΙΑ	9
000010101	0.825671	ΕΠΙΤΥΧΙΑ	10
000010111	0.388705	ΕΠΙΤΥΧΙΑ	11
000011001	0.867537	ΕΠΙΤΥΧΙΑ	12
000011011	0.400661	ΕΠΙΤΥΧΙΑ	13
000011101	0.850132	ΕΠΙΤΥΧΙΑ	14
000011111	0.844144	ΕΠΙΤΥΧΙΑ	15
000100011	0.861831	ΕΠΙΤΥΧΙΑ	16
000100101	0.861831	ΕΠΙΤΥΧΙΑ	17
000100111	0.419005	ΕΠΙΤΥΧΙΑ	18
000101001	0.007460	ΑΠΟΤΥΧΙΑ	19
000101011	0.406722	ΕΠΙΤΥΧΙΑ	20
000101101	0.412837	ΕΠΙΤΥΧΙΑ	21
000101111	0.394656	ΕΠΙΤΥΧΙΑ	22
000110011	0.856028	ΕΠΙΤΥΧΙΑ	23
000110101	0.400661	ΕΠΙΤΥΧΙΑ	24
000110111	0.412837	ΕΠΙΤΥΧΙΑ	25
000111001	0.406722	ΕΠΙΤΥΧΙΑ	26
000111011	0.861831	ΕΠΙΤΥΧΙΑ	27
000111101	0.406722	ΕΠΙΤΥΧΙΑ	28
000111111	0.850132	ΕΠΙΤΥΧΙΑ	29
001000011	0.850132	ΕΠΙΤΥΧΙΑ	30
001000101	0.856028	ΕΠΙΤΥΧΙΑ	31
001000111	0.400661	ΕΠΙΤΥΧΙΑ	32
001001011	0.861831	ΕΠΙΤΥΧΙΑ	33

001001101 0.394656 ΕΠΙΤΥΧΙΑ 34
001001111 0.861831 ΕΠΙΤΥΧΙΑ 35
001010011 0.001503 ΑΠΟΤΥΧΙΑ 36
001010101 0.123541 ΕΠΙΤΥΧΙΑ 37
001010111 0.394656 ΕΠΙΤΥΧΙΑ 38
001011011 0.850132 ΕΠΙΤΥΧΙΑ 39
001011101 0.412837 ΕΠΙΤΥΧΙΑ 40
001011111 0.131006 ΕΠΙΤΥΧΙΑ 41
001100101 0.007110 ΑΠΟΤΥΧΙΑ 42
001100111 0.850132 ΕΠΙΤΥΧΙΑ 43
001101011 0.400661 ΕΠΙΤΥΧΙΑ 44
001101101 0.844144 ΕΠΙΤΥΧΙΑ 45
001101111 0.412837 ΕΠΙΤΥΧΙΑ 46
001110101 0.000009 ΑΠΟΤΥΧΙΑ 47
001110111 0.856028 ΕΠΙΤΥΧΙΑ 48
001111011 0.856028 ΕΠΙΤΥΧΙΑ 49
001111101 0.400661 ΕΠΙΤΥΧΙΑ 50
001111111 0.128476 ΕΠΙΤΥΧΙΑ 51
010000011 0.844144 ΕΠΙΤΥΧΙΑ 52
010000111 0.406722 ΕΠΙΤΥΧΙΑ 53
010001011 0.856028 ΕΠΙΤΥΧΙΑ 54
010001111 0.861831 ΕΠΙΤΥΧΙΑ 55
010010011 0.856028 ΕΠΙΤΥΧΙΑ 56
010010111 0.856028 ΕΠΙΤΥΧΙΑ 57
010011011 0.400661 ΕΠΙΤΥΧΙΑ 58
010011111 0.406722 ΕΠΙΤΥΧΙΑ 59
010100011 0.856028 ΕΠΙΤΥΧΙΑ 60
010100111 0.000000 ΑΠΟΤΥΧΙΑ 61
010101011 0.131006 ΕΠΙΤΥΧΙΑ 62
010101111 0.856028 ΕΠΙΤΥΧΙΑ 63
010110011 0.856028 ΕΠΙΤΥΧΙΑ 64
010110111 0.856028 ΕΠΙΤΥΧΙΑ 65
010111011 0.850132 ΕΠΙΤΥΧΙΑ 66
010111111 0.394656 ΕΠΙΤΥΧΙΑ 67
011000111 0.850132 ΕΠΙΤΥΧΙΑ 68
011001111 0.856028 ΕΠΙΤΥΧΙΑ 69
011010111 0.856028 ΕΠΙΤΥΧΙΑ 70

011011111 0.406722 ΕΠΙΤΥΧΙΑ 71
011101111 0.406722 ΕΠΙΤΥΧΙΑ 72
011111111 0.394656 ΕΠΙΤΥΧΙΑ 73
100000000 0.856028 ΕΠΙΤΥΧΙΑ 74
100010000 0.032849 ΕΠΙΤΥΧΙΑ 75
100100000 0.131006 ΕΠΙΤΥΧΙΑ 76
100101000 0.850132 ΕΠΙΤΥΧΙΑ 77
100110000 0.861831 ΕΠΙΤΥΧΙΑ 78
100111000 0.861831 ΕΠΙΤΥΧΙΑ 79
101000000 0.856028 ΕΠΙΤΥΧΙΑ 80
101000100 0.406722 ΕΠΙΤΥΧΙΑ 81
101001000 0.406722 ΕΠΙΤΥΧΙΑ 82
101001100 0.000282 ΑΠΟΤΥΧΙΑ 83
101010000 0.133579 ΕΠΙΤΥΧΙΑ 84
101010100 0.032849 ΕΠΙΤΥΧΙΑ 85
101011000 0.035109 ΕΠΙΤΥΧΙΑ 86
101011100 0.400661 ΕΠΙΤΥΧΙΑ 87
101100000 0.125988 ΕΠΙΤΥΧΙΑ 88
101100100 0.128476 ΕΠΙΤΥΧΙΑ 89
101101000 0.844144 ΕΠΙΤΥΧΙΑ 90
101101100 0.033587 ΕΠΙΤΥΧΙΑ 91
101110000 0.844144 ΕΠΙΤΥΧΙΑ 92
101110100 0.406722 ΕΠΙΤΥΧΙΑ 93
101111000 0.850132 ΕΠΙΤΥΧΙΑ 94
101111100 0.128476 ΕΠΙΤΥΧΙΑ 95
110000000 0.032849 ΕΠΙΤΥΧΙΑ 96
110000010 0.844144 ΕΠΙΤΥΧΙΑ 97
110000100 0.850132 ΕΠΙΤΥΧΙΑ 98
110001000 0.406722 ΕΠΙΤΥΧΙΑ 99
110001010 0.844144 ΕΠΙΤΥΧΙΑ 100
110010000 0.406722 ΕΠΙΤΥΧΙΑ 101
110010010 0.412837 ΕΠΙΤΥΧΙΑ 102
110010100 0.838070 ΕΠΙΤΥΧΙΑ 103
110011000 0.856028 ΕΠΙΤΥΧΙΑ 104
110011010 0.825671 ΕΠΙΤΥΧΙΑ 105
110100000 0.856028 ΕΠΙΤΥΧΙΑ 106
110100010 0.406722 ΕΠΙΤΥΧΙΑ 107

110100100 0.844144 ΕΠΙΤΥΧΙΑ 108
110101000 0.412837 ΕΠΙΤΥΧΙΑ 109
110101010 0.850132 ΕΠΙΤΥΧΙΑ 110
110101100 0.412837 ΕΠΙΤΥΧΙΑ 111
110110000 0.388705 ΕΠΙΤΥΧΙΑ 112
110110010 0.844144 ΕΠΙΤΥΧΙΑ 113
110110100 0.856028 ΕΠΙΤΥΧΙΑ 114
110111000 0.850132 ΕΠΙΤΥΧΙΑ 115
110111010 0.861831 ΕΠΙΤΥΧΙΑ 116
110111100 0.406722 ΕΠΙΤΥΧΙΑ 117
111000000 0.035109 ΕΠΙΤΥΧΙΑ 118
111000010 0.125988 ΕΠΙΤΥΧΙΑ 119
111000100 0.406722 ΕΠΙΤΥΧΙΑ 120
111000110 0.856028 ΕΠΙΤΥΧΙΑ 121
111001000 0.844144 ΕΠΙΤΥΧΙΑ 122
111001010 0.131006 ΕΠΙΤΥΧΙΑ 123
111001100 0.850132 ΕΠΙΤΥΧΙΑ 124
111010000 0.850132 ΕΠΙΤΥΧΙΑ 125
111010010 0.406722 ΕΠΙΤΥΧΙΑ 126
111010100 0.133579 ΕΠΙΤΥΧΙΑ 127
111010110 0.406722 ΕΠΙΤΥΧΙΑ 128
111011000 0.400661 ΕΠΙΤΥΧΙΑ 129
111011010 0.856028 ΕΠΙΤΥΧΙΑ 130
111011100 0.856028 ΕΠΙΤΥΧΙΑ 131
111100000 0.406722 ΕΠΙΤΥΧΙΑ 132
111100010 0.406722 ΕΠΙΤΥΧΙΑ 133
111100100 0.850132 ΕΠΙΤΥΧΙΑ 134
111100110 0.128476 ΕΠΙΤΥΧΙΑ 135
111101000 0.850132 ΕΠΙΤΥΧΙΑ 136
111101010 0.419005 ΕΠΙΤΥΧΙΑ 137
111101100 0.125988 ΕΠΙΤΥΧΙΑ 138
111101110 0.412837 ΕΠΙΤΥΧΙΑ 139
111110000 0.032126 ΕΠΙΤΥΧΙΑ 140
111110010 0.861831 ΕΠΙΤΥΧΙΑ 141
111110100 0.856028 ΕΠΙΤΥΧΙΑ 142
111110110 0.007641 ΑΠΟΤΥΧΙΑ 143
111111000 0.382811 ΕΠΙΤΥΧΙΑ 144

111111010 0.867537 ΕΠΙΤΥΧΙΑ 145
111111100 0.400661 ΕΠΙΤΥΧΙΑ 146
111111110 0.394656 ΕΠΙΤΥΧΙΑ 147

Όπως βλέπουμε ύστερα από τον αναλυτικό έλεγχο της ακολουθίας που παράχθηκε τα αποτελέσματα είναι συνολικά καλύτερα από αυτά ενός LFSR. Αν εξαιρέσουμε τον έλεγχο Non- Overlapping Template Matching, όπου υστερεί έναντι του LFSR, η ακολουθία επιτυγχάνει σε όλους τους άλλους ελέγχους. Η μη εκτέλεση τριών ελέγχων είναι κάτι που αναμέναμε όπως εξηγείται στα σχόλια. Για την καλύτερη αξιολόγηση της γεννήτριας αλλά και την καλύτερη κατανόηση της συμπεριφοράς της σουίτας ελέγχων που χρησιμοποιείται υποβλήθηκαν σε έλεγχο 10 μόνο ακολουθίες μήκους 204bits μόνο η κάθε μια, που δημιουργήθηκαν από αυτή. Τα συνοπτικά αποτελέσματα όπως τα εμφανίζει η σουίτα είναι:

RESULTS FOR THE UNIFORMITY OF P-VALUES AND THE PROPORTION OF PASSING SEQUENCES

generator is <GEFFE.txt>

C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	P-VALUE		PROPORTION		STATISTICAL TEST
2	1	1	0	1	1	0	2	2	0	0.739918		10/10		Frequency
2	0	1	1	0	2	0	2	2	0	0.534146		10/10		BlockFrequency
2	1	0	2	0	0	0	2	2	1	0.534146		10/10		CumulativeSums
2	1	0	2	0	0	0	2	1	2	0.534146		10/10		CumulativeSums
1	0	2	0	1	2	1	1	2	0	0.739918		10/10		Runs
1	2	1	0	1	1	0	1	2	1	0.911413		9/10		LongestRun
10	0	0	0	0	0	0	0	0	0	0.000000	*	0/10	*	Rank
1	0	2	0	3	2	0	0	0	2	0.213309		10/10		FFT
1	0	0	0	0	0	0	0	0	9	0.000000	*	10/10		NonOverlappingTemplate
1	0	0	0	0	0	0	0	0	9	0.000000	*	10/10		NonOverlappingTemplate
4	0	0	0	0	0	0	0	0	6	0.000003	*	10/10		NonOverlappingTemplate
0	0	0	0	0	0	0	0	0	10	0.000000	*	10/10		NonOverlappingTemplate
1	0	0	0	0	0	0	0	0	9	0.000000	*	10/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	9/10		NonOverlappingTemplate
4	0	0	0	0	0	0	0	0	6	0.000003	*	9/10		NonOverlappingTemplate
0	0	0	0	0	0	0	0	0	10	0.000000	*	10/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	10/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
5	0	0	0	0	0	0	0	0	5	0.000008	*	10/10		NonOverlappingTemplate
5	0	0	0	0	0	0	0	0	5	0.000008	*	9/10		NonOverlappingTemplate
0	0	0	0	0	0	0	0	0	10	0.000000	*	10/10		NonOverlappingTemplate
4	0	0	0	0	0	0	0	0	6	0.000003	*	10/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
1	0	0	0	0	0	0	0	0	9	0.000000	*	10/10		NonOverlappingTemplate
1	0	0	0	0	0	0	0	0	9	0.000000	*	10/10		NonOverlappingTemplate
1	0	0	0	0	0	0	0	0	9	0.000000	*	10/10		NonOverlappingTemplate
1	0	0	0	0	0	0	0	0	9	0.000000	*	10/10		NonOverlappingTemplate
5	0	0	0	0	0	0	0	0	5	0.000008	*	9/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	10/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	9/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	9/10		NonOverlappingTemplate
0	0	0	0	0	0	0	0	0	10	0.000000	*	10/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	10/10		NonOverlappingTemplate
1	0	0	0	0	0	0	0	0	9	0.000000	*	10/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	10/10		NonOverlappingTemplate
1	0	0	0	0	0	0	0	0	9	0.000000	*	10/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
1	0	0	0	0	0	0	0	0	9	0.000000	*	10/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	10/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	10/10		NonOverlappingTemplate
1	0	0	0	0	0	0	0	0	9	0.000000	*	10/10		NonOverlappingTemplate
4	0	0	0	0	0	0	0	0	6	0.000003	*	10/10		NonOverlappingTemplate
1	0	0	0	0	0	0	0	0	9	0.000000	*	10/10		NonOverlappingTemplate
6	0	0	0	0	0	0	0	0	4	0.000003	*	9/10		NonOverlappingTemplate
5	0	0	0	0	0	0	0	0	5	0.000008	*	9/10		NonOverlappingTemplate
5	0	0	0	0	0	0	0	0	5	0.000008	*	10/10		NonOverlappingTemplate
1	0	0	0	0	0	0	0	0	9	0.000000	*	9/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
6	0	0	0	0	0	0	0	0	4	0.000003	*	8/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	10/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	9/10		NonOverlappingTemplate

4	0	0	0	0	0	0	0	0	6	0.000003	*	10/10		NonOverlappingTemplate
1	0	0	0	0	0	0	0	0	9	0.000000	*	10/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	9/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
1	0	0	0	0	0	0	0	0	9	0.000000	*	10/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	10/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	9/10		NonOverlappingTemplate
1	0	0	0	0	0	0	0	0	9	0.000000	*	10/10		NonOverlappingTemplate
1	0	0	0	0	0	0	0	0	9	0.000000	*	10/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	10/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
5	0	0	0	0	0	0	0	0	5	0.000008	*	7/10	*	NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	9/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
1	0	0	0	0	0	0	0	0	9	0.000000	*	9/10		NonOverlappingTemplate
1	0	0	0	0	0	0	0	0	9	0.000000	*	9/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	9/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	10/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	10/10		NonOverlappingTemplate
1	0	0	0	0	0	0	0	0	9	0.000000	*	10/10		NonOverlappingTemplate
4	0	0	0	0	0	0	0	0	6	0.000003	*	9/10		NonOverlappingTemplate
4	0	0	0	0	0	0	0	0	6	0.000003	*	9/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	10/10		NonOverlappingTemplate
0	0	0	0	0	0	0	0	0	10	0.000000	*	10/10		NonOverlappingTemplate
1	0	0	0	0	0	0	0	0	9	0.000000	*	10/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	10/10		NonOverlappingTemplate
5	0	0	0	0	0	0	0	0	5	0.000008	*	10/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	8/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	9/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	8/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	10/10		NonOverlappingTemplate
5	0	0	0	0	0	0	0	0	5	0.000008	*	10/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	10/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	9/10		NonOverlappingTemplate
4	0	0	0	0	0	0	0	0	6	0.000003	*	10/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	9/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
4	0	0	0	0	0	0	0	0	6	0.000003	*	10/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	9/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	10/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	9/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	10/10		NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000	*	9/10		NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000	*	10/10		NonOverlappingTemplate
4	0	0	0	0	0	0	0	0	6	0.000003	*	10/10		NonOverlappingTemplate
0	0	0	0	0	0	0	0	0	10	0.000000	*	10/10		NonOverlappingTemplate

6	0	0	0	0	0	0	0	0	4	0.000003 *	10/10	NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000 *	10/10	NonOverlappingTemplate
1	0	0	0	0	0	0	0	0	9	0.000000 *	10/10	NonOverlappingTemplate
4	0	0	0	0	0	0	0	0	6	0.000003 *	10/10	NonOverlappingTemplate
1	0	0	0	0	0	0	0	0	9	0.000000 *	10/10	NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000 *	10/10	NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000 *	10/10	NonOverlappingTemplate
5	0	0	0	0	0	0	0	0	5	0.000008 *	10/10	NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000 *	10/10	NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000 *	10/10	NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000 *	10/10	NonOverlappingTemplate
1	0	0	0	0	0	0	0	0	9	0.000000 *	10/10	NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000 *	10/10	NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000 *	9/10	NonOverlappingTemplate
5	0	0	0	0	0	0	0	0	5	0.000008 *	10/10	NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000 *	9/10	NonOverlappingTemplate
1	0	0	0	0	0	0	0	0	9	0.000000 *	10/10	NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000 *	10/10	NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000 *	9/10	NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000 *	10/10	NonOverlappingTemplate
2	0	0	0	0	0	0	0	0	8	0.000000 *	10/10	NonOverlappingTemplate
3	0	0	0	0	0	0	0	0	7	0.000000 *	9/10	NonOverlappingTemplate
1	0											

Συμπεράσματα: Η Geffe είναι μια γεννήτρια που ουσιαστικά αποτελεί αναβάθμιση των LFSRs, καθώς εισάγεται η μη γραμμικότητα. Έχει, επίσης, δεδομένο μειονέκτημα ως προς την πιθανή συσχέτιση της εξόδου της, με κάποιο από τις εξόδους των LFSRs που χρησιμοποιούνται ως είσοδοι. Για αυτούς τους λόγους, καθώς και τους εκτενείς ελέγχους σε LFSRs που προηγήθηκαν, κρίθηκε σκόπιμο αρχικά να παρουσιαστούν αναλυτικά τα αποτελέσματα μιας μετρίου μήκους (ως προς τις απαιτήσεις της σουίτας) ακολουθία, γεγονός που κατέστησε δυνατή την απεικόνιση της και στη συνέχεια να γίνει έλεγχος σε πολλαπλές ακολουθίες πραγματικά μικρού μήκους, για να υπάρξει ένα πρώτο δείγμα συμπεριφοράς της σουίτας ως προς μη ενδεδειγμένα (για αυτήν πάντα) δεδομένα. Τα αποτελέσματα δείχνουν ότι η γεννήτρια εμφανίζει αρκετά συχνά ακολουθίες που εμφανίζουν σημαντικό αριθμό μη περιοδικών μοτίβων. Σαφέστατη βελτίωση όλων των υπολοίπων αποτελεσμάτων στους ελέγχους που είναι εκτελέσιμοι. Η συμπεριφορά του ελέγχου Binary Matrix Rank, όπου στην μετρίου μήκους ακολουθία παρουσιάζει ¨πολύ καλό¨ αποτέλεσμα, ενώ στις μικρές αποτυγχάνει πλήρως δεν είναι επιλήψιμο, γιατί για τις μικρές ακολουθίες δεν μπορεί να σχηματιστεί ούτε ένας πίνακας 32χ32 και έτσι οι έλεγχοι δεν εκτελούνται. Η συμπεριφορά του ελέγχου Linear Complexity είναι πολύ ενδιαφέρουσα γιατί αν και στις προδιαγραφές του ζητάει ακολουθίες μεγαλύτερες ή ίσες των 10^6bits, την ακολουθία των 2048 bits την εξέτασε κανονικά (χωρίζοντας την σε 4 υπακολουθίες των 500 bits και απέρριψε τα 4 τελευταία bits) και μάλιστα έδωσε επιτυχία, όπως θα έπρεπε αφού η ακολουθία έχει προκύψει από μια γραμμική σύνθεση LFSRs. Ενώ για τις μικρότερες ακολουθίες (των 204 bits) ο έλεγχος δεν μπόρεσε να εκτελεστεί. Γενικότερα βλέπουμε ότι η σουίτα παρουσιάζει προβλήματα σε μικρού μήκους δεδομένα, ενώ στα μετρίου η συμπεριφορά της ποικίλει ανάλογα την περίσταση. Όμως παρά τους πειραματισμούς μπορούμε να συμπεράνουμε ότι η Geffe είναι μια καλή ΓΨΑ, αλλά επισφαλής για κρυπτογραφικές εφαρμογές.

4.4 Αξιολόγηση ΓΨΑ LCG

Με σκοπό την δημιουργία μιας ψευδοτυχαίας ακολουθίας αριθμών μεγέθους 1000000 bits δόθηκε σε μια ΓΨΑ LCG αρχική τιμή $Z_0 = 23482349$ και ακολουθήθηκε η διαδικασία όπως αυτή αναφέρεται στην 3.9 μέχρι και την κατασκευή των $U_i \in [0,1], \iota = 1,2,\dots,10^6$, τότε: αν $U_i \leq 0.5$ ως έξοδος θεωρήθηκε: ο αριθμός 0, ενώ αν $1 < U_i \leq 1$ ως έξοδος θεωρήθηκε ο αριθμός 0. Η παραγόμενη ακολουθίας των 10^6 bits, η οποία αποτελείται από 499800 το πλήθος άσσους και 500200 το πλήθος μηδενικά, υποβλήθηκε στη σουίτα στατιστικών ελέγχων και τα αποτελέσματα είναι τα ακόλουθα:

Τεστ	P-value
Frequency (Monobit)	0.689157 (ΕΠΙΤΥΧΙΑ)
Frequency within a block ($M = 128$)	0.648667 (ΕΠΙΤΥΧΙΑ)
Runs	0.879061 (ΕΠΙΤΥΧΙΑ)
Longest Run of Ones in a block ($N = 100, M = 10000$)	0.359480 (ΕΠΙΤΥΧΙΑ)
Binary Matrix Rank	0.421067 (ΕΠΙΤΥΧΙΑ)
DFT (Spectral)	0.734207 (ΕΠΙΤΥΧΙΑ)
Non-Overlapping Template Matching	Για $m = 9$ υπολογίζονται 148 P-values δείτε αναλυτικά παρακάτω
Overlapping Template Matching ($m = 9$)	0.957083 (ΕΠΙΤΥΧΙΑ)
Maurer's "Universal Statistical" ($L = 7, Q = 1280, K = 141577$)	0.846210 (ΕΠΙΤΥΧΙΑ)
Linear Complexity (M=500)	0.458514 (ΕΠΙΤΥΧΙΑ)
Serial ($m = 16$) P-value1	0.365401 (ΕΠΙΤΥΧΙΑ)
Serial ($m = 16$) P-value2	0.875408 (ΕΠΙΤΥΧΙΑ)
Approximate Entropy ($m = 10$)	0.383722 (ΕΠΙΤΥΧΙΑ)
Cumulative Sums (Cusum) forward	0.804883 (ΕΠΙΤΥΧΙΑ)
Cumulative Sums (Cusum) reverse	0.980302 (ΕΠΙΤΥΧΙΑ)
Random Excursions	Υπολογίζονται 8 P-values δείτε αναλυτικά παρακάτω
Random Excursions Variant	Υπολογίζονται 18 P-values δείτε αναλυτικά παρακάτω

Για το Non-overlapping Template Matching έχουμε για τα μη περιοδικά μοτίβο που εξετάστηκαν:

Μοτίβο	P-value	Αποτέλεσμα	Α/α
000000001	0.149799	ΕΠΙΤΥΧΙΑ	0
000000011	0.459605	ΕΠΙΤΥΧΙΑ	1
000000101	0.478476	ΕΠΙΤΥΧΙΑ	2
000000111	0.014049	ΕΠΙΤΥΧΙΑ	3
000001001	0.144080	ΕΠΙΤΥΧΙΑ	4
000001011	0.882581	ΕΠΙΤΥΧΙΑ	5
000001101	0.016935	ΕΠΙΤΥΧΙΑ	6
000001111	0.335704	ΕΠΙΤΥΧΙΑ	7
000010001	0.673353	ΕΠΙΤΥΧΙΑ	8
000010011	0.053040	ΕΠΙΤΥΧΙΑ	9
000010101	0.737846	ΕΠΙΤΥΧΙΑ	10
000010111	0.184338	ΕΠΙΤΥΧΙΑ	11
000011001	0.044231	ΕΠΙΤΥΧΙΑ	12
000011011	0.092521	ΕΠΙΤΥΧΙΑ	13
000011101	0.170511	ΕΠΙΤΥΧΙΑ	14
000011111	0.809764	ΕΠΙΤΥΧΙΑ	15
000100011	0.594157	ΕΠΙΤΥΧΙΑ	16
000100101	0.850280	ΕΠΙΤΥΧΙΑ	17
000100111	0.436746	ΕΠΙΤΥΧΙΑ	18
000101001	0.273746	ΕΠΙΤΥΧΙΑ	19
000101011	0.709515	ΕΠΙΤΥΧΙΑ	20
000101101	0.228003	ΕΠΙΤΥΧΙΑ	21
000101111	0.319493	ΕΠΙΤΥΧΙΑ	22
000110011	0.166371	ΕΠΙΤΥΧΙΑ	23
000110101	0.822540	ΕΠΙΤΥΧΙΑ	24
000110111	0.014125	ΕΠΙΤΥΧΙΑ	25
000111001	0.686946	ΕΠΙΤΥΧΙΑ	26
000111011	0.406305	ΕΠΙΤΥΧΙΑ	27
000111101	0.956759	ΕΠΙΤΥΧΙΑ	28
000111111	0.425708	ΕΠΙΤΥΧΙΑ	29
001000011	0.148305	ΕΠΙΤΥΧΙΑ	30
001000101	0.346760	ΕΠΙΤΥΧΙΑ	31
001000111	0.005512	ΑΠΟΤΥΧΙΑ	32
001001011	0.203688	ΕΠΙΤΥΧΙΑ	33
001001101	0.791827	ΕΠΙΤΥΧΙΑ	34
001001111	0.681514	ΕΠΙΤΥΧΙΑ	35

001010011 0.483826 ΕΠΙΤΥΧΙΑ 36
001010101 0.619648 ΕΠΙΤΥΧΙΑ 37
001010111 0.525756 ΕΠΙΤΥΧΙΑ 38
001011011 0.060153 ΕΠΙΤΥΧΙΑ 39
001011101 0.924490 ΕΠΙΤΥΧΙΑ 40
001011111 0.031024 ΕΠΙΤΥΧΙΑ 41
001100101 0.686002 ΕΠΙΤΥΧΙΑ 42
001100111 0.122757 ΕΠΙΤΥΧΙΑ 43
001101011 0.681750 ΕΠΙΤΥΧΙΑ 44
001101101 0.011462 ΕΠΙΤΥΧΙΑ 45
001101111 0.354386 ΕΠΙΤΥΧΙΑ 46
001110101 0.620832 ΕΠΙΤΥΧΙΑ 47
001110111 0.549192 ΕΠΙΤΥΧΙΑ 48
001111011 0.767175 ΕΠΙΤΥΧΙΑ 49
001111101 0.631732 ΕΠΙΤΥΧΙΑ 50
001111111 0.033861 ΕΠΙΤΥΧΙΑ 51
010000011 0.002491 ΑΠΟΤΥΧΙΑ 52
010000111 0.974554 ΕΠΙΤΥΧΙΑ 53
010001011 0.023730 ΕΠΙΤΥΧΙΑ 54
010001111 0.175645 ΕΠΙΤΥΧΙΑ 55
010010011 0.475323 ΕΠΙΤΥΧΙΑ 56
010010111 0.051878 ΕΠΙΤΥΧΙΑ 57
010011011 0.779869 ΕΠΙΤΥΧΙΑ 58
010011111 0.270108 ΕΠΙΤΥΧΙΑ 59
010100011 0.325410 ΕΠΙΤΥΧΙΑ 60
010100111 0.089543 ΕΠΙΤΥΧΙΑ 61
010101011 0.093699 ΕΠΙΤΥΧΙΑ 62
010101111 0.062538 ΕΠΙΤΥΧΙΑ 63
010110011 0.585234 ΕΠΙΤΥΧΙΑ 64
010110111 0.318812 ΕΠΙΤΥΧΙΑ 65
010111011 0.501263 ΕΠΙΤΥΧΙΑ 66
010111111 0.183280 ΕΠΙΤΥΧΙΑ 67
011000111 0.795734 ΕΠΙΤΥΧΙΑ 68
011001111 0.690484 ΕΠΙΤΥΧΙΑ 69
011010111 0.493945 ΕΠΙΤΥΧΙΑ 70
011011111 0.705768 ΕΠΙΤΥΧΙΑ 71
011101111 0.733235 ΕΠΙΤΥΧΙΑ 72

011111111 0.003376 ΑΠΟΤΥΧΙΑ 73
100000000 0.149799 ΕΠΙΤΥΧΙΑ 74
100010000 0.372609 ΕΠΙΤΥΧΙΑ 75
100100000 0.754423 ΕΠΙΤΥΧΙΑ 76
100101000 0.327746 ΕΠΙΤΥΧΙΑ 77
100110000 0.234630 ΕΠΙΤΥΧΙΑ 78
100111000 0.956547 ΕΠΙΤΥΧΙΑ 79
101000000 0.671459 ΕΠΙΤΥΧΙΑ 80
101000100 0.136890 ΕΠΙΤΥΧΙΑ 81
101001000 0.203628 ΕΠΙΤΥΧΙΑ 82
101001100 0.227078 ΕΠΙΤΥΧΙΑ 83
101010000 0.900499 ΕΠΙΤΥΧΙΑ 84
101010100 0.881346 ΕΠΙΤΥΧΙΑ 85
101011000 0.060594 ΕΠΙΤΥΧΙΑ 86
101011100 0.219269 ΕΠΙΤΥΧΙΑ 87
101100000 0.931393 ΕΠΙΤΥΧΙΑ 88
101100100 0.753742 ΕΠΙΤΥΧΙΑ 89
101101000 0.578912 ΕΠΙΤΥΧΙΑ 90
101101100 0.594275 ΕΠΙΤΥΧΙΑ 91
101110000 0.327572 ΕΠΙΤΥΧΙΑ 92
101110100 0.111933 ΕΠΙΤΥΧΙΑ 93
101111000 0.089993 ΕΠΙΤΥΧΙΑ 94
101111100 0.628057 ΕΠΙΤΥΧΙΑ 95
110000000 0.783729 ΕΠΙΤΥΧΙΑ 96
110000010 0.076186 ΕΠΙΤΥΧΙΑ 97
110000100 0.547005 ΕΠΙΤΥΧΙΑ 98
110001000 0.117369 ΕΠΙΤΥΧΙΑ 99
110001010 0.495383 ΕΠΙΤΥΧΙΑ 100
110010000 0.463885 ΕΠΙΤΥΧΙΑ 101
110010010 0.267778 ΕΠΙΤΥΧΙΑ 102
110010100 0.195127 ΕΠΙΤΥΧΙΑ 103
110011000 0.612197 ΕΠΙΤΥΧΙΑ 104
110011010 0.967387 ΕΠΙΤΥΧΙΑ 105
110100000 0.917699 ΕΠΙΤΥΧΙΑ 106
110100010 0.251747 ΕΠΙΤΥΧΙΑ 107
110100100 0.441017 ΕΠΙΤΥΧΙΑ 108
110101000 0.459498 ΕΠΙΤΥΧΙΑ 109

110101010 0.476517 ΕΠΙΤΥΧΙΑ 110
110101100 0.133200 ΕΠΙΤΥΧΙΑ 111
110110000 0.173512 ΕΠΙΤΥΧΙΑ 112
110110010 0.090054 ΕΠΙΤΥΧΙΑ 113
110110100 0.556001 ΕΠΙΤΥΧΙΑ 114
110111000 0.572141 ΕΠΙΤΥΧΙΑ 115
110111010 0.357504 ΕΠΙΤΥΧΙΑ 116
110111100 0.606646 ΕΠΙΤΥΧΙΑ 117
111000000 0.086652 ΕΠΙΤΥΧΙΑ 118
111000010 0.438202 ΕΠΙΤΥΧΙΑ 119
111000100 0.063060 ΕΠΙΤΥΧΙΑ 120
111000110 0.944472 ΕΠΙΤΥΧΙΑ 121
111001000 0.723159 ΕΠΙΤΥΧΙΑ 122
111001010 0.295696 ΕΠΙΤΥΧΙΑ 123
111001100 0.670156 ΕΠΙΤΥΧΙΑ 124
111010000 0.813143 ΕΠΙΤΥΧΙΑ 125
111010010 0.130301 ΕΠΙΤΥΧΙΑ 126
111010100 0.382422 ΕΠΙΤΥΧΙΑ 127
111010110 0.520096 ΕΠΙΤΥΧΙΑ 128
111011000 0.024382 ΕΠΙΤΥΧΙΑ 129
111011010 0.232471 ΕΠΙΤΥΧΙΑ 130
111011100 0.974632 ΕΠΙΤΥΧΙΑ 131
111100000 0.152920 ΕΠΙΤΥΧΙΑ 132
111100010 0.085495 ΕΠΙΤΥΧΙΑ 133
111100100 0.379834 ΕΠΙΤΥΧΙΑ 134
111100110 0.341380 ΕΠΙΤΥΧΙΑ 135
111101000 0.404413 ΕΠΙΤΥΧΙΑ 136
111101010 0.840827 ΕΠΙΤΥΧΙΑ 137
111101100 0.261761 ΕΠΙΤΥΧΙΑ 138
111101110 0.573890 ΕΠΙΤΥΧΙΑ 139
111110000 0.179538 ΕΠΙΤΥΧΙΑ 140
111110010 0.854443 ΕΠΙΤΥΧΙΑ 141
111110100 0.174097 ΕΠΙΤΥΧΙΑ 142
111110110 0.789537 ΕΠΙΤΥΧΙΑ 143
111111000 0.441540 ΕΠΙΤΥΧΙΑ 144
111111010 0.279968 ΕΠΙΤΥΧΙΑ 145
111111100 0.739686 ΕΠΙΤΥΧΙΑ 146

111111110 0.002776 ΑΠΟΤΥΧΙΑ 147

Για το Random Excursions test (μετά από 1918 κύκλους J) έχουμε:

ΕΠΙΤΥΧΙΑ x = -4 P-value = 0.397490
ΕΠΙΤΥΧΙΑ x = -3 P-value = 0.181590
ΕΠΙΤΥΧΙΑ x = -2 P-value = 0.530058
ΕΠΙΤΥΧΙΑ x = -1 P-value = 0.516009
ΕΠΙΤΥΧΙΑ x = 1 P-value = 0.852023
ΕΠΙΤΥΧΙΑ x = 2 P-value = 0.230713
ΕΠΙΤΥΧΙΑ x = 3 P-value = 0.189886
ΕΠΙΤΥΧΙΑ x = 4 P-value = 0.023180

Για το Random Excursions Variant test (μετά από 1918 κύκλους J) έχουμε:

ΕΠΙΤΥΧΙΑ (x = -9) P-value = 0.227775
ΕΠΙΤΥΧΙΑ (x = -8) P-value = 0.285871
ΕΠΙΤΥΧΙΑ (x = -7) P-value = 0.276522
ΕΠΙΤΥΧΙΑ (x = -6) P-value = 0.232988
ΕΠΙΤΥΧΙΑ (x = -5) P-value = 0.227989
ΕΠΙΤΥΧΙΑ (x = -4) P-value = 0.155057
ΕΠΙΤΥΧΙΑ (x = -3) P-value = 0.288493
ΕΠΙΤΥΧΙΑ (x = -2) P-value = 0.412034
ΕΠΙΤΥΧΙΑ (x = -1) P-value = 0.419498
ΕΠΙΤΥΧΙΑ (x = 1) P-value = 0.734563
ΕΠΙΤΥΧΙΑ (x = 2) P-value = 0.925730
ΕΠΙΤΥΧΙΑ (x = 3) P-value = 0.515784
ΕΠΙΤΥΧΙΑ (x = 4) P-value = 0.185419
ΕΠΙΤΥΧΙΑ (x = 5) P-value = 0.143224
ΕΠΙΤΥΧΙΑ (x = 6) P-value = 0.121605
ΕΠΙΤΥΧΙΑ (x = 7) P-value = 0.102156
ΕΠΙΤΥΧΙΑ (x = 8) P-value = 0.064777
ΕΠΙΤΥΧΙΑ (x = 9) P-value = 0.081405

Όπως βλέπουμε η γεννήτρια παρουσιάζει πολύ καλά αποτελέσματα στους ελέγχους με βάση την ακολουθία που εξετάστηκε, με μοναδικό ψεγάδι κάποια μη περιοδικά μοτίβο που εμφανίζονται. Η αξιολόγηση της συνεχίστηκε υποβάλλοντας στα τεστ 100 ακολουθίες μεγέθους 1000000bits, που παρήχθησαν από αυτήν. Για να μπορέσουν έτσι να

παρουσιαστούν σημαντικά ευρήματα για την αξιολόγηση της γεννήτριας. Ακολουθεί η συνοπτική παρουσίαση των αποτελεσμάτων της σουίτας:

```
-------------------------------------------------------------------------------
RESULTS FOR THE UNIFORMITY OF P-VALUES AND THE PROPORTION OF PASSING SEQUENCES
-------------------------------------------------------------------------------
   generator is <Linear-Congruential>
-------------------------------------------------------------------------------
 C1  C2  C3  C4  C5  C6  C7  C8  C9 C10  P-VALUE  PROPORTION  STATISTICAL TEST
-------------------------------------------------------------------------------
  6  15   8   9  11   7  10  11  13  10  0.678686     99/100     Frequency
 11   7  11   6  10  13  15  13   6   8  0.437274     99/100     BlockFrequency
  6  12  17  10   3  11  10  15   9   7  0.080519     99/100     CumulativeSums
  7  13  11  10   9  10  11  13  11   5  0.779188     99/100     CumulativeSums
 12  10  11   8   9   8  14  11   9   8  0.935716     97/100     Runs
  8   8  10  18   7   9  13  10  11   6  0.289667    100/100     LongestRun
 12   9   9   8   8   7  18  10  10   9  0.455937     99/100     Rank
  5  11  12  11  13   5  15  13   8   7  0.262249    100/100     FFT
 13  13  10   9  12   7  11   9   8   8  0.897763     99/100     NonOverlappingTemplate
 10  13   7   9  10  12  10  13   6  10  0.851383    100/100     NonOverlappingTemplate
  6  15  11   6  10   8   8  13  14   9  0.419021    100/100     NonOverlappingTemplate
 14   7  10   6  14  17   9   5  11   7  0.115387     97/100     NonOverlappingTemplate
 12   6  13   7  11  11  15  12   6   7  0.401199     99/100     NonOverlappingTemplate
  7  12  12   7   8  11  10  12  15   6  0.574903    100/100     NonOverlappingTemplate
 13   9   9  12   9   8   7   9  11  13  0.911413     99/100     NonOverlappingTemplate
 18  10  13   3   8   9  11   9  12   7  0.115387     99/100     NonOverlappingTemplate
  5   9   9   6  13  13  12  14   6  13  0.304126    100/100     NonOverlappingTemplate
 19   9   9   8  15  13   8   7   7   5  0.051942     98/100     NonOverlappingTemplate
 10  13  11   9   6  13   5  10  10  13  0.637119     97/100     NonOverlappingTemplate
 14  11  10   9   6  10  14  13   7   6  0.494392     98/100     NonOverlappingTemplate
 14  10  11  10   7   9  13  12   7   7  0.759756     98/100     NonOverlappingTemplate
 15   9   8  10  11   6   9  10   8  14  0.657933    100/100     NonOverlappingTemplate
 12  13   4   8  14   6   6   9  17  11  0.085587     99/100     NonOverlappingTemplate
12  13   8  13  10   9  10   8   7  10  0.911413     98/100     NonOverlappingTemplate
10  10   7  11   8  16  12   8   8  10  0.719747     99/100     NonOverlappingTemplate
 6  15  11   9  12  12   5   4  14  12  0.153763    100/100     NonOverlappingTemplate
10  11  14  15   9  12   8   9   7   5  0.474986     99/100     NonOverlappingTemplate
 6  12  16   8  11  15   6   8  10   8  0.275709    100/100     NonOverlappingTemplate
14  10   9   5  10  10   9  14  13   6  0.494392     97/100     NonOverlappingTemplate
11   7  11  15  16   8  14   5   7   6  0.115387     98/100     NonOverlappingTemplate
15   7   9  12  14   8  11  10   7   7  0.554420     96/100     NonOverlappingTemplate
 8  14  11   8   6  14   9   8  12  10  0.678686     98/100     NonOverlappingTemplate
 8  14  11   9  15   9   8   8   6  12  0.574903     99/100     NonOverlappingTemplate
12   8  13  10   6   9  11  11  12   8  0.883171    100/100     NonOverlappingTemplate
 9  11  12   6   9   8   7  11  16  11  0.595549     99/100     NonOverlappingTemplate
 8  13   5  10   9   7  14  11  12  11  0.637119    100/100     NonOverlappingTemplate
 8  14   9  13  11  14   6   9   6  10  0.534146     99/100     NonOverlappingTemplate
 9   5   8   9  13   9  18   8  13   8  0.202268    100/100     NonOverlappingTemplate
10  11   3   9  15  14  11  12  10   5  0.202268     99/100     NonOverlappingTemplate
13   8  11  12   6   7  11  11  10  11  0.867692    100/100     NonOverlappingTemplate
 7   9  11  10  12  10  11  10  10  10  0.996335     99/100     NonOverlappingTemplate
 9  14   9   7   4  14   9  11   9  14  0.366918    100/100     NonOverlappingTemplate
11  15  12   8  13   5  14  12   2   8  0.075719     99/100     NonOverlappingTemplate
10   9   3  10   9  11  13  10  19   6  0.071177    100/100     NonOverlappingTemplate
14  18   5   6  16   9  12   5   7   8  0.017912    100/100     NonOverlappingTemplate
 7   9  11  12  11   8  11  12   6  13  0.834308    100/100     NonOverlappingTemplate
11   8   6  14   9  13   8  12  12   7  0.657933     98/100     NonOverlappingTemplate
 7  12   8   8  14  10  13   9  11   8  0.816537     98/100     NonOverlappingTemplate
13   5   8  12   9  11   8  15   6  13  0.366918     97/100     NonOverlappingTemplate
11   8  10  11  11   8  14   9  11   7  0.924076    100/100     NonOverlappingTemplate
13   9  12  10   7  13   4  11   8  13  0.514124    100/100     NonOverlappingTemplate
10  13  14   9  10   8   9   6  12   9  0.816537     99/100     NonOverlappingTemplate
10   6  10  13  12  11  12  16   2   8  0.129620     98/100     NonOverlappingTemplate
```

9	13	8	13	4	8	11	11	10	13	0.595549	100/100	NonOverlappingTemplate
13	5	8	14	11	12	8	12	4	13	0.262249	99/100	NonOverlappingTemplate
8	10	11	12	10	8	12	8	10	11	0.987896	99/100	NonOverlappingTemplate
10	10	5	9	8	12	13	13	11	9	0.798139	99/100	NonOverlappingTemplate
11	9	14	11	13	11	6	11	8	6	0.678686	99/100	NonOverlappingTemplate
16	9	9	6	11	12	17	5	9	6	0.090936	98/100	NonOverlappingTemplate
11	8	10	11	11	10	10	15	6	8	0.816537	99/100	NonOverlappingTemplate
14	9	9	14	8	7	9	8	14	8	0.616305	98/100	NonOverlappingTemplate
11	18	9	9	10	7	11	9	8	8	0.474986	99/100	NonOverlappingTemplate
9	4	15	9	16	6	13	6	14	8	0.066882	100/100	NonOverlappingTemplate
10	7	11	11	10	7	18	7	10	9	0.401199	100/100	NonOverlappingTemplate
13	6	11	9	11	12	13	7	10	8	0.798139	96/100	NonOverlappingTemplate
10	12	13	10	9	10	5	9	9	13	0.834308	99/100	NonOverlappingTemplate
15	14	6	6	11	10	7	9	11	11	0.474986	99/100	NonOverlappingTemplate
12	12	9	8	9	12	12	9	10	7	0.955835	100/100	NonOverlappingTemplate
10	7	11	9	8	10	12	9	14	10	0.935716	98/100	NonOverlappingTemplate
13	11	11	10	9	6	7	13	11	9	0.851383	98/100	NonOverlappingTemplate
8	9	8	13	8	11	16	10	9	8	0.699313	99/100	NonOverlappingTemplate
12	8	8	13	4	10	9	12	10	14	0.554420	99/100	NonOverlappingTemplate
7	15	9	12	10	11	7	9	11	9	0.816537	98/100	NonOverlappingTemplate
10	7	13	14	7	13	8	11	9	8	0.719747	100/100	NonOverlappingTemplate
11	9	13	8	7	16	8	11	8	9	0.637119	99/100	NonOverlappingTemplate
7	13	10	12	11	11	8	11	5	12	0.759756	100/100	NonOverlappingTemplate
11	5	10	19	11	5	11	13	6	9	0.066882	99/100	NonOverlappingTemplate
15	15	10	11	3	6	11	13	8	8	0.145326	98/100	NonOverlappingTemplate
15	10	7	9	14	11	10	16	5	3	0.062821	100/100	NonOverlappingTemplate
10	14	13	9	8	9	13	9	6	9	0.759756	98/100	NonOverlappingTemplate
7	11	6	10	7	7	15	13	13	11	0.455937	99/100	NonOverlappingTemplate
14	15	8	13	9	6	7	6	9	13	0.304126	98/100	NonOverlappingTemplate
13	13	10	9	12	7	11	9	8	8	0.897763	99/100	NonOverlappingTemplate
17	8	9	11	8	8	8	10	11	10	0.657933	100/100	NonOverlappingTemplate
7	9	9	10	16	14	11	12	7	5	0.334538	100/100	NonOverlappingTemplate
13	9	12	13	14	10	6	8	9	6	0.574903	99/100	NonOverlappingTemplate
15	10	9	6	10	12	8	7	12	11	0.699313	99/100	NonOverlappingTemplate
8	9	13	13	13	8	6	15	6	9	0.401199	99/100	NonOverlappingTemplate
10	12	10	11	8	14	8	8	7	12	0.867692	99/100	NonOverlappingTemplate
9	17	9	10	8	11	7	8	10	11	0.637119	100/100	NonOverlappingTemplate
13	10	5	13	7	14	5	13	7	13	0.213309	97/100	NonOverlappingTemplate
12	9	12	9	14	8	14	7	6	9	0.616305	99/100	NonOverlappingTemplate
9	12	11	6	14	12	11	9	5	11	0.637119	100/100	NonOverlappingTemplate
11	12	9	15	9	5	14	10	7	8	0.474986	100/100	NonOverlappingTemplate
10	12	15	8	12	12	7	8	6	10	0.637119	99/100	NonOverlappingTemplate
9	10	14	10	10	14	12	8	3	10	0.437274	99/100	NonOverlappingTemplate
11	5	8	17	12	7	8	15	9	8	0.181557	99/100	NonOverlappingTemplate
7	13	11	11	15	6	6	10	6	15	0.224821	100/100	NonOverlappingTemplate
10	11	12	7	10	13	8	12	9	8	0.935716	100/100	NonOverlappingTemplate
11	11	14	11	3	14	9	10	3	14	0.090936	99/100	NonOverlappingTemplate
7	10	14	8	10	8	12	6	9	16	0.437274	100/100	NonOverlappingTemplate
5	5	10	10	10	10	16	12	15	7	0.191687	100/100	NonOverlappingTemplate
12	13	6	8	9	9	12	10	8	13	0.816537	100/100	NonOverlappingTemplate
11	12	11	8	9	8	14	10	9	8	0.935716	98/100	NonOverlappingTemplate
12	10	12	13	8	17	6	8	6	8	0.275709	99/100	NonOverlappingTemplate
10	15	11	4	9	12	7	9	16	7	0.202268	99/100	NonOverlappingTemplate
9	11	8	15	5	17	10	13	6	6	0.102526	99/100	NonOverlappingTemplate
12	13	8	10	15	10	5	9	6	12	0.455937	98/100	NonOverlappingTemplate
15	12	9	4	7	11	9	17	5	11	0.085587	97/100	NonOverlappingTemplate
8	7	10	11	20	11	7	8	9	9	0.162606	97/100	NonOverlappingTemplate
6	7	17	12	16	7	10	8	11	6	0.108791	99/100	NonOverlappingTemplate
11	6	13	10	13	11	10	8	12	6	0.739918	100/100	NonOverlappingTemplate
10	8	7	12	8	10	9	13	10	13	0.911413	99/100	NonOverlappingTemplate

8	8	6	8	13	8	20	9	7	13	0.066882	99/100	NonOverlappingTemplate
9	12	10	8	13	3	10	10	12	13	0.534146	100/100	NonOverlappingTemplate
11	10	8	15	7	9	7	17	6	10	0.249284	99/100	NonOverlappingTemplate
12	10	13	9	9	10	13	8	6	10	0.883171	99/100	NonOverlappingTemplate
13	7	7	9	13	11	9	10	14	7	0.699313	97/100	NonOverlappingTemplate
7	10	11	17	9	7	5	10	12	12	0.334538	100/100	NonOverlappingTemplate
12	14	11	9	11	9	8	9	9	8	0.946308	100/100	NonOverlappingTemplate
10	9	9	8	13	6	17	11	6	11	0.366918	100/100	NonOverlappingTemplate
15	7	11	6	16	12	8	7	11	7	0.249284	100/100	NonOverlappingTemplate
10	12	10	9	10	5	9	13	7	15	0.595549	99/100	NonOverlappingTemplate
12	10	10	4	13	12	8	12	8	11	0.678686	100/100	NonOverlappingTemplate
5	6	10	17	10	12	13	7	10	10	0.262249	99/100	NonOverlappingTemplate
10	6	13	11	4	7	12	15	10	12	0.319084	99/100	NonOverlappingTemplate
10	14	10	8	7	8	13	14	8	8	0.678686	100/100	NonOverlappingTemplate
8	8	8	10	14	7	13	11	8	13	0.739918	100/100	NonOverlappingTemplate
10	10	10	9	11	10	12	11	11	6	0.983453	98/100	NonOverlappingTemplate
10	10	13	10	11	5	13	7	14	7	0.554420	99/100	NonOverlappingTemplate
9	15	11	15	10	9	4	10	10	7	0.366918	100/100	NonOverlappingTemplate
8	8	11	6	7	9	13	12	7	19	0.129620	100/100	NonOverlappingTemplate
7	10	7	12	9	11	12	17	6	9	0.401199	98/100	NonOverlappingTemplate
9	9	15	10	7	11	9	10	9	11	0.911413	100/100	NonOverlappingTemplate
8	10	12	11	11	7	12	10	10	9	0.983453	99/100	NonOverlappingTemplate
11	12	11	12	9	9	12	7	11	6	0.897763	97/100	NonOverlappingTemplate
11	11	8	10	10	12	8	11	10	9	0.996335	100/100	NonOverlappingTemplate
12	8	8	10	14	6	12	10	8	12	0.779188	97/100	NonOverlappingTemplate
9	6	8	12	13	9	8	13	11	11	0.834308	99/100	NonOverlappingTemplate
16	8	7	8	6	11	13	6	11	14	0.262249	98/100	NonOverlappingTemplate
11	15	11	6	10	10	9	9	12	7	0.759756	100/100	NonOverlappingTemplate
11	8	8	9	9	12	10	13	11	9	0.978072	98/100	NonOverlappingTemplate
11	6	13	10	8	13	6	6	17	10	0.213309	99/100	NonOverlappingTemplate
5	8	11	9	11	13	12	6	13	12	0.595549	99/100	NonOverlappingTemplate
12	8	8	6	15	10	8	18	8	7	0.145326	100/100	NonOverlappingTemplate
9	16	11	7	5	11	10	7	10	14	0.366918	98/100	NonOverlappingTemplate
5	10	12	11	10	6	13	13	9	11	0.678686	100/100	NonOverlappingTemplate
6	8	16	10	8	12	10	6	14	10	0.383827	98/100	NonOverlappingTemplate
12	12	13	7	11	5	10	8	15	7	0.437274	100/100	NonOverlappingTemplate
7	11	6	12	14	5	14	6	11	14	0.213309	99/100	NonOverlappingTemplate
8	17	9	13	9	8	7	11	5	13	0.262249	100/100	NonOverlappingTemplate
13	10	9	6	12	9	11	6	10	14	0.699313	98/100	NonOverlappingTemplate
13	8	11	13	8	11	9	12	5	10	0.759756	100/100	NonOverlappingTemplate
11	9	13	13	6	11	10	9	11	7	0.851383	98/100	NonOverlappingTemplate
18	7	9	10	16	9	6	6	10	9	0.108791	98/100	NonOverlappingTemplate
14	15	8	13	9	7	6	6	9	13	0.304126	98/100	NonOverlappingTemplate
12	16	13	10	10	8	7	10	7	7	0.534146	99/100	OverlappingTemplate
8	9	8	9	8	12	9	15	15	7	0.554420	99/100	Universal
10	9	8	14	14	14	7	9	5	10	0.455937	99/100	ApproximateEntropy
5	2	5	3	9	8	7	4	7	6	0.383827	56/56	RandomExcursions
5	5	8	3	6	3	8	2	8	8	0.289667	56/56	RandomExcursions
1	4	4	5	7	5	10	6	7	7	0.262249	56/56	RandomExcursions
5	4	8	3	7	5	7	4	7	6	0.779188	55/56	RandomExcursions
3	7	5	4	2	9	4	8	3	11	0.051942	56/56	RandomExcursions
4	4	8	5	4	8	7	2	8	6	0.455937	55/56	RandomExcursions
6	6	4	6	6	1	7	4	11	5	0.191687	55/56	RandomExcursions
5	11	5	7	7	1	3	1	7	9	0.017912	56/56	RandomExcursions
6	10	13	4	3	2	4	5	4	5	0.011791	54/56	RandomExcursionsVariant
6	10	8	8	6	2	1	7	3	5	0.075719	54/56	RandomExcursionsVariant
7	8	8	5	6	5	4	5	4	4	0.816537	55/56	RandomExcursionsVariant
5	7	8	8	4	6	5	2	3	8	0.419021	56/56	RandomExcursionsVariant
5	6	7	8	6	8	2	3	4	7	0.494392	56/56	RandomExcursionsVariant
3	7	7	5	7	5	9	4	3	6	0.574903	56/56	RandomExcursionsVariant

```
  3   4   6   9   6   7   9   4   3   5  0.383827     56/56      RandomExcursionsVariant
  2   6   8   3   9   7   6   5   2   8  0.191687     55/56      RandomExcursionsVariant
  4   2  11   5   5   7   4   6   6   6  0.289667     56/56      RandomExcursionsVariant
  4   7   2  10   3   4   9   5   6   6  0.191687     56/56      RandomExcursionsVariant
  6   4   3   8   8   5   3   4   8   7  0.494392     56/56      RandomExcursionsVariant
  5   4   1   5   9   8   8   5   4   7  0.262249     55/56      RandomExcursionsVariant
  4   3   5   5   8   7   8   8   2   6  0.419021     56/56      RandomExcursionsVariant
  4   6   4   4   4   9   8   4   6   7  0.616305     56/56      RandomExcursionsVariant
  7   3   6   3   9   5   5   4   7   7  0.574903     56/56      RandomExcursionsVariant
  5   7   4   6   3  10   6   5   9   1  0.137282     56/56      RandomExcursionsVariant
  6   7   4   3   5   6   7   7   5   6  0.911413     56/56      RandomExcursionsVariant
  7   3   5   5   6   7   3   6  10   4  0.455937     56/56      RandomExcursionsVariant
  7   8  11  17  12   9  14   6   9   7  0.275709    100/100     Serial
 13   8   7   9  17   8  14   8   9   7  0.304126     99/100     Serial
  9  12   6  10  15  13   7  11   9   8  0.637119     99/100     LinearComplexity

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
The minimum pass rate for each statistical test with the exception of the
random excursion (variant) test is approximately = 96 for a
sample size = 100 binary sequences.

The minimum pass rate for the random excursion (variant) test
is approximately = 53 for a sample size = 56 binary sequences.

For further guidelines construct a probability table using the MAPLE program
provided in the addendum section of the documentation.
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

Συμπεράσματα: Λόγω της κακής συμπεριφοράς που είδαμε ότι παρουσιάζουν αρκετοί έλεγχοι της σουίτας για ακολουθίες μικρού μήκους, και της εναλλάσουσας που παρουσιάζουν κάποιοι (λίγοι) για ακολουθίες μετρίου μήκους. Σε αυτό τα παράδειγμα και οι 100 ακολουθίες που εξετάστηκαν ήταν μεγέθους 1000000 bits. Όπως αναφέρθηκε προηγουμένως η βασικά παρουσιαζόμενη ακολουθία πέρασε επιτυχώς σχεδόν όλα τα τεστ με μοναδική εξαίρεση κάποια μη περιοδικά μοτίβο που εμφάνιζε. Στην καλύτερη εικόνα του δείγματος 100 ακολουθιών, βλέπουμε ότι σε όποιο τεστ (όλα πλην των Random Excursions και Random Excursions Variant) έχουν υποβληθεί 100 έλεγχοι, έχει επιτευχθεί το minimum των (περίπου) 96 επιτυχιών και έχουμε ότι η γεννήτρια επιτυγχάνει σε αυτά. Για τα Random Excursions και Random Excursions Variant έχουμε ότι από τις 100 ακολουθίες οι 44 δεν σχημάτιζαν επαρκή αριθμό κύκλων (J) για να υποβληθούν στα tests. Οι υπόλοιπες 56 όμως που υποβλήθεισαν έπιασαν την αναλογία των (περίπου) 53 επιτυχιών και θεωρείται ότι η γεννήτρια περνάει και αυτά τα tests. Επίσης πρέπει να τονιστεί ότι ενώ η γεννήτρια είναι εντελώς γραμμική είναι φυσιολογικό το ότι περνάει το Linear Complexity test, γιατί αυτό έχει ως κύριο σκοπό να ελέγξει την πολυπλοκότητα ενός LFSR και όχι τη γραμμικότητα μιας γεννήτρια που δεν χρησιμοποιεί LFSRs.

4.5 Αξιολόγηση ΓΨΑ BBS

Ακολουθήθηκε η διαδικασία όπως αυτή περιγράφεται στην 3.10 με σκοπό τη δημιουργία μιας ακολουθίας μεγέθους 1000000 bits από την ΓΨΑ BBS. Η παραγόμενη ακολουθίας των 10^6 bits, η οποία αποτελείται από 499614 το πλήθος άσσους και 500386 το πλήθος μηδενικά, υποβλήθηκε στη σουίτα στατιστικών ελέγχων και τα αποτελέσματα είναι τα ακόλουθα:

Τεστ	P-value
Frequency (Monobit)	0.440114 (ΕΠΙΤΥΧΙΑ)
Frequency within a block ($M = 128$)	0.690945 (ΕΠΙΤΥΧΙΑ)
Runs	0.876503 (ΕΠΙΤΥΧΙΑ)
Longest Run of Ones in a block ($N = 100, M = 10000$)	0.323696 (ΕΠΙΤΥΧΙΑ)
Binary Matrix Rank	0.375644 (ΕΠΙΤΥΧΙΑ)
DFT (Spectral)	0.183317 (ΕΠΙΤΥΧΙΑ)
Non-Overlapping Template Matching	Για $m = 9$ υπολογίζονται 148 P-values δείτε αναλυτικά παρακάτω
Overlapping Template Matching ($m = 9$)	0.664788 (ΕΠΙΤΥΧΙΑ)
Maurer's "Universal Statistical" ($L = 7, Q = 1280, K = 141577$)	0.943714 (ΕΠΙΤΥΧΙΑ)
Linear Complexity (M=500)	0.402085 (ΕΠΙΤΥΧΙΑ)
Serial ($m = 16$) P-value1	0.173651 (ΕΠΙΤΥΧΙΑ)
Serial ($m = 16$) P-value2	0.701539 (ΕΠΙΤΥΧΙΑ)
Approximate Entropy (m=10)	0.710534 (ΕΠΙΤΥΧΙΑ)
Cumulative Sums (Cusum) forward	0.241797 (ΕΠΙΤΥΧΙΑ)
Cumulative Sums (Cusum) reverse	0.362976 (ΕΠΙΤΥΧΙΑ)
Random Excursions	ΜΗ ΕΦΑΡΜΟΣΙΜΟ
Random Excursions Variant	ΜΗ ΕΦΑΡΜΟΣΙΜΟ

Σχόλιο:

Τα τεστ Random Excursions και Random Excursions Variant δεν μπορούν να εφαρμοστούν γιατί δημιουργούνται μόνο 86 κύκλοι (J) με αποτέλεσμα να έχουμε 8 στο πρώτο και 18 στο δεύτερο (τεστ) P-values ίσες με 0.000000. Εδώ θα πρέπει να τονιστεί ότι το μήκος της ακολουθίας ήταν εντός των συνιστώμενων από των κατασκευαστή τιμών, όμως δημιουργήθηκε ανεπαρκής αριθμός κύκλων και δεν μπόρεσαν να εφαρμοστούν τα δυο tests.

Για το Non-overlapping Template Matching έχουμε για τα μη περιοδικά μοτίβο που εξετάστηκαν:

Μοτίβο	P-value	Αποτέλεσμα	Α/α
000000001	0.666839	ΕΠΙΤΥΧΙΑ	0
000000011	0.407202	ΕΠΙΤΥΧΙΑ	1
000000101	0.197593	ΕΠΙΤΥΧΙΑ	2
000000111	0.199846	ΕΠΙΤΥΧΙΑ	3
000001001	0.115632	ΕΠΙΤΥΧΙΑ	4
000001011	0.945741	ΕΠΙΤΥΧΙΑ	5
000001101	0.036818	ΕΠΙΤΥΧΙΑ	6
000001111	0.752264	ΕΠΙΤΥΧΙΑ	7
000010001	0.439348	ΕΠΙΤΥΧΙΑ	8
000010011	0.775547	ΕΠΙΤΥΧΙΑ	9
000010101	0.801983	ΕΠΙΤΥΧΙΑ	10
000010111	0.326101	ΕΠΙΤΥΧΙΑ	11
000011001	0.678322	ΕΠΙΤΥΧΙΑ	12
000011011	0.427451	ΕΠΙΤΥΧΙΑ	13
000011101	0.784388	ΕΠΙΤΥΧΙΑ	14
000011111	0.259778	ΕΠΙΤΥΧΙΑ	15
000100011	0.182063	ΕΠΙΤΥΧΙΑ	16
000100101	0.583125	ΕΠΙΤΥΧΙΑ	17
000100111	0.531096	ΕΠΙΤΥΧΙΑ	18
000101001	0.064896	ΕΠΙΤΥΧΙΑ	19
000101011	0.705650	ΕΠΙΤΥΧΙΑ	20
000101101	0.486676	ΕΠΙΤΥΧΙΑ	21
000101111	0.850864	ΕΠΙΤΥΧΙΑ	22
000110011	0.117103	ΕΠΙΤΥΧΙΑ	23
000110101	0.163120	ΕΠΙΤΥΧΙΑ	24
000110111	0.024256	ΕΠΙΤΥΧΙΑ	25
000111001	0.851835	ΕΠΙΤΥΧΙΑ	26
000111011	0.805512	ΕΠΙΤΥΧΙΑ	27
000111101	0.013337	ΕΠΙΤΥΧΙΑ	28
000111111	0.781415	ΕΠΙΤΥΧΙΑ	29
001000011	0.923054	ΕΠΙΤΥΧΙΑ	30
001000101	0.923486	ΕΠΙΤΥΧΙΑ	31
001000111	0.415537	ΕΠΙΤΥΧΙΑ	32

001001011 0.745304 ΕΠΙΤΥΧΙΑ 33
001001101 0.533147 ΕΠΙΤΥΧΙΑ 34
001001111 0.276268 ΕΠΙΤΥΧΙΑ 35
001010011 0.071429 ΕΠΙΤΥΧΙΑ 36
001010101 0.132227 ΕΠΙΤΥΧΙΑ 37
001010111 0.328440 ΕΠΙΤΥΧΙΑ 38
001011011 0.884074 ΕΠΙΤΥΧΙΑ 39
001011101 0.255486 ΕΠΙΤΥΧΙΑ 40
001011111 0.141827 ΕΠΙΤΥΧΙΑ 41
001100101 0.896042 ΕΠΙΤΥΧΙΑ 42
001100111 0.146824 ΕΠΙΤΥΧΙΑ 43
001101011 0.280433 ΕΠΙΤΥΧΙΑ 44
001101101 0.611488 ΕΠΙΤΥΧΙΑ 45
001101111 0.357688 ΕΠΙΤΥΧΙΑ 46
001110101 0.694961 ΕΠΙΤΥΧΙΑ 47
001110111 0.243280 ΕΠΙΤΥΧΙΑ 48
001111011 0.987906 ΕΠΙΤΥΧΙΑ 49
001111101 0.312979 ΕΠΙΤΥΧΙΑ 50
001111111 0.733004 ΕΠΙΤΥΧΙΑ 51
010000011 0.748047 ΕΠΙΤΥΧΙΑ 52
010000111 0.552766 ΕΠΙΤΥΧΙΑ 53
010001011 0.275273 ΕΠΙΤΥΧΙΑ 54
010001111 0.153638 ΕΠΙΤΥΧΙΑ 55
010010011 0.994385 ΕΠΙΤΥΧΙΑ 56
010010111 0.476952 ΕΠΙΤΥΧΙΑ 57
010011011 0.192802 ΕΠΙΤΥΧΙΑ 58
010011111 0.082888 ΕΠΙΤΥΧΙΑ 59
010100011 0.733351 ΕΠΙΤΥΧΙΑ 60
010100111 0.058644 ΕΠΙΤΥΧΙΑ 61
010101011 0.191532 ΕΠΙΤΥΧΙΑ 62
010101111 0.280666 ΕΠΙΤΥΧΙΑ 63
010110011 0.516938 ΕΠΙΤΥΧΙΑ 64
010110111 0.380887 ΕΠΙΤΥΧΙΑ 65
010111011 0.471531 ΕΠΙΤΥΧΙΑ 66
010111111 0.125752 ΕΠΙΤΥΧΙΑ 67
011000111 0.668142 ΕΠΙΤΥΧΙΑ 68
011001111 0.721415 ΕΠΙΤΥΧΙΑ 69

011010111 0.042833 ΕΠΙΤΥΧΙΑ 70
011011111 0.118975 ΕΠΙΤΥΧΙΑ 71
011101111 0.130551 ΕΠΙΤΥΧΙΑ 72
011111111 0.978295 ΕΠΙΤΥΧΙΑ 73
100000000 0.666839 ΕΠΙΤΥΧΙΑ 74
100010000 0.745419 ΕΠΙΤΥΧΙΑ 75
100100000 0.388212 ΕΠΙΤΥΧΙΑ 76
100101000 0.357781 ΕΠΙΤΥΧΙΑ 77
100110000 0.441226 ΕΠΙΤΥΧΙΑ 78
100111000 0.116611 ΕΠΙΤΥΧΙΑ 79
101000000 0.273594 ΕΠΙΤΥΧΙΑ 80
101000100 0.538058 ΕΠΙΤΥΧΙΑ 81
101001000 0.727104 ΕΠΙΤΥΧΙΑ 82
101001100 0.456302 ΕΠΙΤΥΧΙΑ 83
101010000 0.781415 ΕΠΙΤΥΧΙΑ 84
101010100 0.629953 ΕΠΙΤΥΧΙΑ 85
101011000 0.325928 ΕΠΙΤΥΧΙΑ 86
101011100 0.163322 ΕΠΙΤΥΧΙΑ 87
101100000 0.671577 ΕΠΙΤΥΧΙΑ 88
101100100 0.593217 ΕΠΙΤΥΧΙΑ 89
101101000 0.311136 ΕΠΙΤΥΧΙΑ 90
101101100 0.118284 ΕΠΙΤΥΧΙΑ 91
101110000 0.007239 ΑΠΟΤΥΧΙΑ 92
101110100 0.916427 ΕΠΙΤΥΧΙΑ 93
101111000 0.282610 ΕΠΙΤΥΧΙΑ 94
101111100 0.351286 ΕΠΙΤΥΧΙΑ 95
110000000 0.390641 ΕΠΙΤΥΧΙΑ 96
110000010 0.543558 ΕΠΙΤΥΧΙΑ 97
110000100 0.959718 ΕΠΙΤΥΧΙΑ 98
110001000 0.814930 ΕΠΙΤΥΧΙΑ 99
110001010 0.668853 ΕΠΙΤΥΧΙΑ 100
110010000 0.283157 ΕΠΙΤΥΧΙΑ 101
110010010 0.799189 ΕΠΙΤΥΧΙΑ 102
110010100 0.677021 ΕΠΙΤΥΧΙΑ 103
110011000 0.772430 ΕΠΙΤΥΧΙΑ 104
110011010 0.425094 ΕΠΙΤΥΧΙΑ 105
110100000 0.680687 ΕΠΙΤΥΧΙΑ 106

110100010 0.708462 ΕΠΙΤΥΧΙΑ 107
110100100 0.551151 ΕΠΙΤΥΧΙΑ 108
110101000 0.563996 ΕΠΙΤΥΧΙΑ 109
110101010 0.630428 ΕΠΙΤΥΧΙΑ 110
110101100 0.226550 ΕΠΙΤΥΧΙΑ 111
110110000 0.118246 ΕΠΙΤΥΧΙΑ 112
110110010 0.822437 ΕΠΙΤΥΧΙΑ 113
110110100 0.648694 ΕΠΙΤΥΧΙΑ 114
110111000 0.297879 ΕΠΙΤΥΧΙΑ 115
110111010 0.200861 ΕΠΙΤΥΧΙΑ 116
110111100 0.319152 ΕΠΙΤΥΧΙΑ 117
111000000 0.696608 ΕΠΙΤΥΧΙΑ 118
111000010 0.542410 ΕΠΙΤΥΧΙΑ 119
111000100 0.400548 ΕΠΙΤΥΧΙΑ 120
111000110 0.826459 ΕΠΙΤΥΧΙΑ 121
111001000 0.052817 ΕΠΙΤΥΧΙΑ 122
111001010 0.821401 ΕΠΙΤΥΧΙΑ 123
111001100 0.922114 ΕΠΙΤΥΧΙΑ 124
111010000 0.607826 ΕΠΙΤΥΧΙΑ 125
111010010 0.435812 ΕΠΙΤΥΧΙΑ 126
111010100 0.783068 ΕΠΙΤΥΧΙΑ 127
111010110 0.641694 ΕΠΙΤΥΧΙΑ 128
111011000 0.863192 ΕΠΙΤΥΧΙΑ 129
111011010 0.169832 ΕΠΙΤΥΧΙΑ 130
111011100 0.246922 ΕΠΙΤΥΧΙΑ 131
111100000 0.787460 ΕΠΙΤΥΧΙΑ 132
111100010 0.911549 ΕΠΙΤΥΧΙΑ 133
111100100 0.413016 ΕΠΙΤΥΧΙΑ 134
111100110 0.992051 ΕΠΙΤΥΧΙΑ 135
111101000 0.787460 ΕΠΙΤΥΧΙΑ 136
111101010 0.817339 ΕΠΙΤΥΧΙΑ 137
111101100 0.962460 ΕΠΙΤΥΧΙΑ 138
111101110 0.770980 ΕΠΙΤΥΧΙΑ 139
111110000 0.852125 ΕΠΙΤΥΧΙΑ 140
111110010 0.747020 ΕΠΙΤΥΧΙΑ 141
111110100 0.511653 ΕΠΙΤΥΧΙΑ 142
111110110 0.668972 ΕΠΙΤΥΧΙΑ 143

111111000 0.901071 ΕΠΙΤΥΧΙΑ 144

111111010 0.090024 ΕΠΙΤΥΧΙΑ 145

111111100 0.964715 ΕΠΙΤΥΧΙΑ 146

111111110 0.978295 ΕΠΙΤΥΧΙΑ 147

Όπως βλέπουμε η γεννήτρια πέρασε με επιτυχία (σχεδόν) όλα τα tests τα οποία ήταν εφαρμόσιμα για την ακολουθία που εξετάστηκε, με μοναδική εξαίρεση την εμφάνιση ενός μόνο απεριοδικού μοτίβο. Η αξιολόγηση της συνεχίστηκε υποβάλλοντας στα tests 100 ακολουθίες μεγέθους 1000000bits, που παρήχθησαν από αυτήν. Για να μπορέσουν έτσι να παρουσιαστούν σημαντικά ευρήματα για την αξιολόγηση της γεννήτριας. Ακολουθεί συνοπτική παρουσίαση των αποτελεσμάτων της σουίτας:

RESULTS FOR THE UNIFORMITY OF P-VALUES AND THE PROPORTION OF PASSING SEQUENCES

generator is <Blum-Blum-Shub>

C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	P-VALUE	PROPORTION	STATISTICAL TEST
8	10	11	9	11	10	6	11	12	12	0.955835	100/100	Frequency
6	12	7	7	10	9	12	14	11	12	0.699313	99/100	BlockFrequency
4	15	8	11	15	6	10	11	9	11	0.275709	100/100	CumulativeSums
8	10	8	12	11	17	9	9	6	10	0.534146	100/100	CumulativeSums
11	9	11	15	4	10	9	8	10	13	0.554420	99/100	Runs
16	10	9	14	10	11	9	7	6	8	0.494392	98/100	LongestRun
13	8	6	6	14	8	12	10	13	10	0.554420	98/100	Rank
7	11	9	9	8	14	9	10	13	10	0.897763	99/100	FFT
11	6	14	13	6	6	16	12	7	9	0.191687	99/100	NonOverlappingTemplate
9	10	9	11	11	14	6	10	9	11	0.924076	99/100	NonOverlappingTemplate
14	7	11	7	11	11	7	13	8	11	0.739918	99/100	NonOverlappingTemplate
13	10	12	7	15	7	7	11	12	6	0.474986	98/100	NonOverlappingTemplate
13	14	12	3	8	10	9	13	10	8	0.383827	98/100	NonOverlappingTemplate
12	9	8	10	6	5	19	10	12	9	0.137282	100/100	NonOverlappingTemplate
11	14	10	7	8	9	10	7	14	10	0.779188	99/100	NonOverlappingTemplate
4	9	8	13	14	13	13	9	8	9	0.437274	98/100	NonOverlappingTemplate
6	8	17	13	16	11	10	6	8	5	0.066882	100/100	NonOverlappingTemplate
12	10	11	9	8	9	11	7	12	11	0.978072	100/100	NonOverlappingTemplate
13	9	12	4	11	4	13	10	15	9	0.202268	96/100	NonOverlappingTemplate
16	10	11	11	12	8	7	9	6	10	0.616305	99/100	NonOverlappingTemplate
13	16	6	10	3	9	10	8	11	14	0.153763	100/100	NonOverlappingTemplate
7	8	13	6	13	13	13	8	8	11	0.595549	100/100	NonOverlappingTemplate
6	9	15	12	15	6	7	7	9	14	0.202268	100/100	NonOverlappingTemplate
10	11	13	16	6	11	8	11	11	3	0.224821	99/100	NonOverlappingTemplate
11	6	11	9	12	9	8	12	10	12	0.935716	99/100	NonOverlappingTemplate
17	9	10	8	14	12	7	9	5	9	0.275709	98/100	NonOverlappingTemplate
11	7	13	9	4	10	15	5	8	18	0.042808	98/100	NonOverlappingTemplate
13	5	10	11	7	12	10	9	9	14	0.678686	99/100	NonOverlappingTemplate
11	7	12	12	7	9	7	12	10	13	0.834308	100/100	NonOverlappingTemplate
8	10	11	6	13	10	9	8	15	10	0.739918	100/100	NonOverlappingTemplate
12	12	7	11	11	8	9	10	10	10	0.983453	99/100	NonOverlappingTemplate
10	13	13	10	9	9	7	11	7	11	0.911413	100/100	NonOverlappingTemplate
8	8	11	12	10	12	7	14	8	10	0.867692	100/100	NonOverlappingTemplate
11	9	8	7	12	13	10	11	11	8	0.946308	100/100	NonOverlappingTemplate
11	10	7	9	14	10	7	7	14	11	0.719747	98/100	NonOverlappingTemplate
6	15	9	11	10	10	6	10	10	13	0.657933	100/100	NonOverlappingTemplate
9	17	11	10	6	7	9	13	8	10	0.437274	100/100	NonOverlappingTemplate
15	12	17	10	6	8	6	9	10	7	0.191687	98/100	NonOverlappingTemplate
10	7	14	7	11	11	11	8	6	15	0.514124	99/100	NonOverlappingTemplate
9	12	8	9	11	14	11	8	8	10	0.935716	100/100	NonOverlappingTemplate
4	17	6	11	9	7	11	9	19	7	0.015598	100/100	NonOverlappingTemplate
11	14	18	5	7	15	8	10	8	4	0.030806	97/100	NonOverlappingTemplate
13	10	12	14	9	11	6	6	9	10	0.699313	98/100	NonOverlappingTemplate

7	8	20	10	17	5	9	5	9	10	0.010988	100/100	NonOverlappingTemplate
14	9	6	8	16	11	8	8	10	10	0.514124	99/100	NonOverlappingTemplate
10	13	13	7	11	8	10	9	7	12	0.867692	98/100	NonOverlappingTemplate
8	9	11	6	10	14	16	12	8	6	0.366918	100/100	NonOverlappingTemplate
6	9	13	14	10	7	11	11	10	9	0.798139	99/100	NonOverlappingTemplate
10	9	12	10	9	10	7	12	9	12	0.983453	98/100	NonOverlappingTemplate
10	14	7	11	6	5	7	13	12	15	0.249284	99/100	NonOverlappingTemplate
10	16	13	9	8	12	7	8	10	7	0.574903	98/100	NonOverlappingTemplate
5	15	9	13	8	10	11	11	10	8	0.637119	99/100	NonOverlappingTemplate
9	10	13	13	7	9	16	6	6	11	0.366918	97/100	NonOverlappingTemplate
13	5	8	10	11	8	14	10	9	12	0.699313	98/100	NonOverlappingTemplate
7	14	10	9	8	11	7	15	12	7	0.554420	100/100	NonOverlappingTemplate
12	11	9	8	10	7	10	7	14	12	0.851383	98/100	NonOverlappingTemplate
11	5	13	5	11	16	10	15	6	8	0.115387	99/100	NonOverlappingTemplate
8	12	10	7	8	11	13	6	16	9	0.494392	100/100	NonOverlappingTemplate
13	14	13	9	5	11	9	6	7	13	0.383827	99/100	NonOverlappingTemplate
12	17	5	14	7	10	5	12	6	12	0.085587	99/100	NonOverlappingTemplate
6	10	8	15	10	12	13	12	9	5	0.455937	99/100	NonOverlappingTemplate
10	13	11	6	9	14	5	12	10	10	0.616305	100/100	NonOverlappingTemplate
11	11	8	9	14	11	17	4	5	10	0.145326	100/100	NonOverlappingTemplate
10	7	7	16	10	11	8	12	10	9	0.699313	100/100	NonOverlappingTemplate
7	12	6	13	11	9	7	14	10	11	0.678686	100/100	NonOverlappingTemplate
12	11	12	10	9	9	10	11	6	10	0.971699	99/100	NonOverlappingTemplate
18	11	11	10	5	8	10	8	11	8	0.319084	98/100	NonOverlappingTemplate
12	11	17	10	8	6	7	14	5	10	0.191687	98/100	NonOverlappingTemplate
8	11	8	11	11	7	12	12	12	8	0.935716	100/100	NonOverlappingTemplate
8	13	13	7	9	16	7	11	6	10	0.401199	99/100	NonOverlappingTemplate
10	14	8	10	12	7	15	3	11	10	0.289667	99/100	NonOverlappingTemplate
6	10	8	6	11	13	12	11	8	15	0.534146	100/100	NonOverlappingTemplate
9	7	9	11	12	12	3	17	11	9	0.213309	99/100	NonOverlappingTemplate
11	5	11	7	12	12	12	8	10	12	0.779188	99/100	NonOverlappingTemplate
15	8	8	15	10	7	6	9	9	13	0.401199	99/100	NonOverlappingTemplate
10	11	9	10	10	10	12	10	10	8	0.999438	99/100	NonOverlappingTemplate
5	16	14	11	9	6	7	15	7	10	0.129620	98/100	NonOverlappingTemplate
13	11	10	6	8	8	13	18	9	4	0.108791	100/100	NonOverlappingTemplate
11	7	7	12	13	11	12	8	11	8	0.867692	98/100	NonOverlappingTemplate
8	12	14	8	12	12	6	12	9	7	0.678686	99/100	NonOverlappingTemplate
6	8	15	13	10	13	8	10	10	7	0.574903	99/100	NonOverlappingTemplate
12	6	12	11	5	7	15	14	11	7	0.275709	100/100	NonOverlappingTemplate
11	6	14	13	6	6	16	12	7	9	0.191687	99/100	NonOverlappingTemplate
10	15	8	7	6	8	15	11	10	10	0.494392	100/100	NonOverlappingTemplate
13	7	11	11	13	5	11	9	9	11	0.759756	100/100	NonOverlappingTemplate
15	7	7	11	4	13	11	13	12	7	0.262249	99/100	NonOverlappingTemplate
10	12	11	10	8	13	11	5	12	8	0.816537	100/100	NonOverlappingTemplate
8	11	10	10	9	8	8	14	10	12	0.946308	99/100	NonOverlappingTemplate
11	10	13	8	9	10	9	11	9	10	0.994250	98/100	NonOverlappingTemplate
7	14	10	12	11	9	10	9	9	9	0.946308	98/100	NonOverlappingTemplate
13	8	11	5	6	13	10	13	9	12	0.554420	100/100	NonOverlappingTemplate
15	8	7	5	13	10	12	5	14	11	0.224821	98/100	NonOverlappingTemplate
8	9	2	15	11	11	11	12	13	8	0.249284	100/100	NonOverlappingTemplate
10	11	7	9	7	11	13	8	13	11	0.883171	99/100	NonOverlappingTemplate
6	13	15	11	11	12	9	8	6	9	0.554420	100/100	NonOverlappingTemplate
13	11	10	11	8	5	9	13	8	12	0.759756	99/100	NonOverlappingTemplate
8	3	5	8	18	9	14	15	12	8	0.020548	100/100	NonOverlappingTemplate
10	10	11	13	12	11	6	2	14	11	0.262249	98/100	NonOverlappingTemplate
6	10	8	10	8	10	15	18	8	7	0.181557	98/100	NonOverlappingTemplate
13	10	11	15	10	12	8	6	6	9	0.574903	100/100	NonOverlappingTemplate
10	7	6	8	9	17	11	9	12	11	0.474986	97/100	NonOverlappingTemplate
8	10	9	10	9	11	11	5	12	15	0.719747	100/100	NonOverlappingTemplate
8	11	11	6	11	7	6	17	8	15	0.181557	98/100	NonOverlappingTemplate
7	9	9	12	10	15	11	10	9	8	0.867692	100/100	NonOverlappingTemplate
6	12	15	9	6	7	12	10	14	9	0.419021	99/100	NonOverlappingTemplate
13	7	4	14	12	12	10	4	11	13	0.191687	99/100	NonOverlappingTemplate
15	8	8	6	11	8	13	6	13	12	0.419021	98/100	NonOverlappingTemplate
15	12	7	8	13	7	9	11	8	10	0.678686	99/100	NonOverlappingTemplate
17	9	12	8	11	8	6	8	10	11	0.494392	98/100	NonOverlappingTemplate
14	8	11	12	8	8	4	15	11	9	0.383827	100/100	NonOverlappingTemplate
11	13	8	12	13	6	13	6	11	7	0.554420	100/100	NonOverlappingTemplate
11	10	12	9	8	12	10	16	5	7	0.494392	97/100	NonOverlappingTemplate
8	6	14	8	9	10	7	12	11	15	0.534146	100/100	NonOverlappingTemplate
14	9	4	5	14	13	10	10	9	12	0.289667	99/100	NonOverlappingTemplate
10	9	11	9	14	13	10	9	3	12	0.514124	99/100	NonOverlappingTemplate
5	8	6	14	13	6	12	15	13	8	0.171867	98/100	NonOverlappingTemplate
8	10	10	14	9	6	15	10	6	12	0.514124	100/100	NonOverlappingTemplate
7	13	7	8	19	10	10	7	11	8	0.181557	98/100	NonOverlappingTemplate

14	12	4	8	7	14	14	10	8	9	0.304126	99/100	NonOverlappingTemplate
7	9	10	13	8	9	9	11	12	12	0.946308	98/100	NonOverlappingTemplate
15	10	12	10	12	5	6	9	11	10	0.574903	99/100	NonOverlappingTemplate
11	6	7	9	11	10	11	15	8	12	0.719747	99/100	NonOverlappingTemplate
11	11	3	10	9	12	13	10	9	12	0.637119	98/100	NonOverlappingTemplate
7	8	15	8	8	11	14	15	6	8	0.289667	99/100	NonOverlappingTemplate
5	8	13	16	10	5	7	7	14	15	0.071177	100/100	NonOverlappingTemplate
6	11	8	9	10	16	9	11	9	11	0.719747	98/100	NonOverlappingTemplate
9	10	3	10	12	8	9	18	13	8	0.137282	98/100	NonOverlappingTemplate
7	7	13	11	7	11	12	8	12	12	0.798139	98/100	NonOverlappingTemplate
8	12	9	10	6	16	14	4	10	11	0.249284	100/100	NonOverlappingTemplate
11	11	15	12	7	8	6	8	14	8	0.494392	98/100	NonOverlappingTemplate
16	5	9	13	16	5	12	15	4	5	0.008266	97/100	NonOverlappingTemplate
11	13	11	11	9	9	6	9	8	13	0.883171	99/100	NonOverlappingTemplate
12	10	7	6	12	10	14	7	10	12	0.719747	100/100	NonOverlappingTemplate
7	13	11	9	7	6	15	10	8	14	0.437274	99/100	NonOverlappingTemplate
9	17	8	9	9	13	11	9	7	8	0.534146	100/100	NonOverlappingTemplate
13	8	7	14	10	11	8	9	11	9	0.867692	97/100	NonOverlappingTemplate
7	12	8	8	11	4	18	9	11	12	0.171867	99/100	NonOverlappingTemplate
10	8	8	13	14	6	8	5	15	13	0.262249	99/100	NonOverlappingTemplate
9	8	7	14	9	13	7	11	11	11	0.816537	100/100	NonOverlappingTemplate
10	11	15	6	7	13	4	15	7	12	0.145326	99/100	NonOverlappingTemplate
7	13	19	10	7	12	10	11	8	3	0.055361	99/100	NonOverlappingTemplate
7	12	11	10	13	12	14	6	4	11	0.383827	100/100	NonOverlappingTemplate
11	5	9	10	19	15	6	7	11	7	0.051942	100/100	NonOverlappingTemplate
4	16	12	6	9	3	16	12	10	12	0.028817	100/100	NonOverlappingTemplate
10	11	10	7	9	13	11	11	10	8	0.978072	100/100	NonOverlappingTemplate
15	10	12	12	8	5	12	10	7	9	0.574903	98/100	NonOverlappingTemplate
11	7	9	8	9	14	10	8	12	12	0.883171	99/100	NonOverlappingTemplate
14	12	10	9	14	5	9	10	11	6	0.534146	99/100	NonOverlappingTemplate
7	7	9	18	5	13	9	8	13	11	0.153763	98/100	NonOverlappingTemplate
11	3	16	11	12	5	9	14	7	12	0.102526	99/100	NonOverlappingTemplate
10	14	8	7	16	8	10	6	10	11	0.474986	99/100	NonOverlappingTemplate
11	10	10	3	13	11	7	8	16	11	0.275709	100/100	NonOverlappingTemplate
12	10	12	9	7	6	11	7	10	16	0.534146	98/100	NonOverlappingTemplate
11	16	14	9	7	8	12	10	7	6	0.383827	98/100	NonOverlappingTemplate
14	8	6	8	6	6	11	16	18	7	0.032923	98/100	NonOverlappingTemplate
12	6	12	11	5	7	15	14	11	7	0.275709	100/100	NonOverlappingTemplate
14	4	13	14	11	11	10	7	8	8	0.383827	98/100	OverlappingTemplate
7	12	13	13	13	9	7	11	6	9	0.657933	99/100	Universal
12	12	5	12	9	11	8	15	11	5	0.401199	100/100	ApproximateEntropy
8	6	6	4	13	2	5	7	5	7	0.170294	62/63	RandomExcursions
7	5	9	1	6	9	3	7	8	8	0.311542	62/63	RandomExcursions
4	5	3	4	13	8	6	9	4	7	0.116519	63/63	RandomExcursions
10	3	3	10	10	5	6	8	5	3	0.141256	63/63	RandomExcursions
6	3	10	9	11	5	3	6	3	7	0.155209	62/63	RandomExcursions
6	6	8	10	3	8	2	5	10	5	0.264458	63/63	RandomExcursions
5	4	8	3	6	4	5	11	10	7	0.287306	62/63	RandomExcursions
5	7	6	3	5	5	6	7	7	12	0.484646	63/63	RandomExcursions
2	9	4	6	8	5	5	5	9	10	0.337162	63/63	RandomExcursionsVariant

```
  3   6   2   6   8  11   2   6   8  11  0.057146     63/63      RandomExcursionsVariant
  2   6   5   2   8   8   8   8   7   9  0.364146     63/63      RandomExcursionsVariant
  3   4   4   6  13   3   6   7  11   6  0.051391     63/63      RandomExcursionsVariant
  3   3   6   6   9   7  10   5   6   8  0.517442     63/63      RandomExcursionsVariant
  1   5   7   6   7  10   9   9   3   6  0.222869     63/63      RandomExcursionsVariant
  2   8   3   4   9  10   5   8   8   6  0.264458     63/63      RandomExcursionsVariant
  4   6   2   9   8   8   4  10   6   6  0.392456     63/63      RandomExcursionsVariant
  2   8   7   7   5   5   6   8   5  10  0.585209     63/63      RandomExcursionsVariant
  6   6   5   3   7   6   6   8   9   7  0.900104     63/63      RandomExcursionsVariant
  6   8   5   5   8   6   6   8   7   4  0.957319     63/63      RandomExcursionsVariant
  9   7  10   3   5  10   6   3   6   4  0.287306     63/63      RandomExcursionsVariant
  8   8   5   8   5   9   8   3   6   3  0.585209     63/63      RandomExcursionsVariant
  9   6   4   6   9   6   9   8   3   3  0.452799     63/63      RandomExcursionsVariant
  6   6   6  14   7   4   3   8   4   5  0.105618     63/63      RandomExcursionsVariant
  6   9   8   8   8   1   6   8   5   4  0.422034     61/63      RandomExcursionsVariant
  9   6   8   4   8   3   7   7   6   5  0.788728     62/63      RandomExcursionsVariant
  7   7   7   4   6   8   6   8   5   5  0.970538     62/63      RandomExcursionsVariant
  9  12   9  10  12   4  11  14  10   9  0.699313     99/100     Serial
  5   8   7   9  19  12   9  13  13   5  0.051942     99/100     Serial
 13   7  11   9  13   7   9   9  11  11  0.897763    100/100     LinearComplexity

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
The minimum pass rate for each statistical test with the exception of the
random excursion (variant) test is approximately = 96 for a

 The minimum pass rate for the random excursion (variant) test
 is approximately = 60 for a sample size = 63 binary sequences.

 For further guidelines construct a probability table using the MAPLE program
 provided in the addendum section of the documentation.
 - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

Συμπεράσματα: Σε αυτό τα παράδειγμα και οι 100 ακολουθίες που εξετάστηκαν ήταν μεγέθους 1000000 bits. Όπως αναφέρθηκε προηγουμένως η βασικά παρουσιαζόμενη ακολουθία πέρασε επιτυχώς (σχεδόν) όλα τα τεστ με μοναδική εξαίρεση ένα μη περιοδικά μοτίβο που εμφάνιζε. Στην καλύτερη εικόνα του δείγματος 100 ακολουθιών, βλέπουμε ότι σε όποιο τεστ (όλα πλην των Random Excursions και Random Excursions Variant) έχουν υποβληθεί 100 έλεγχοι, έχει επιτευχθεί όχι απλά το minimum των (περίπου) 96 επιτυχιών, αλλά η αναλογίες κατά μέσο όρο είναι πολύ κοντά στην 99/100, όπως θα αναμέναμε από μια τυχαία γεννήτρια. Προφανέστατα η γεννήτρια επιτυγχάνει σε αυτά τα tests. Για τα Random Excursions και Random Excursions Variant έχουμε ότι από τις 100 ακολουθίες οι 37 δεν σχημάτιζαν επαρκή αριθμό κύκλων (J) για να υποβληθούν στα tests. Οι υπόλοιπες 63 όμως που υπεβλήθησαν όχι απλά έπιασαν την αναλογία των (περίπου) 60/63 επιτυχιών που ζητείται αλλά έδωσαν πολύ καλύτερη αναλογία. Φυσικά θεωρείται ότι η γεννήτρια περνάει και αυτά τα tests. Όπως γίνεται σαφές το δείγμα μιας μόνο ακολουθίας μπορεί να μας δώσει μια πρώτη και αρκετά κατατοπιστική εικόνα για την συμπεριφορά της γεννήτριας, αλλά το δείγμα των 100 ακολουθιών μας παρέχει μια ¨ισχυρή χαρτογράφηση¨ της συμπεριφοράς της. Η ΓΨΑ BBS επιτυγχάνει συντριπτικά τα καλύτερα ποσοστά στα tests ανάμεσα στις γεννήτριες που εξετάστηκαν. Η ασφάλεια της ενδείκνυται για κρυπτογραφικές εφαρμογές. Έχει όμως ένα σαφέστατο μειονέκτημα. Αυτό είναι ο χρόνος που χρειάζεται για να δημιουργήσει/γεννήσει ακολουθίες, είναι δηλαδή πολύ αργή. Ως χαρακτηριστικό παράδειγμα αναφέρεται ότι ο χρόνος που χρειάστηκε για να δημιουργηθούν 100 ακολουθίες μεγέθους 10^6 bits η κάθε μια για την παρούσα εργασία, υπερέβη τις 11 ώρες συνεχόμενης λειτουργίας της γεννήτριας!! Για αυτόν το λόγο δεν ενδείκνυται η χρήση της για κρυπτογράφηση

μεγάλων δεδομένων, ή για εφαρμογές που απαιτούν ταχύτητα (πχ σε κρυπτογραφικούς αλγόριθμους ροής). Λόγω όμως της ασφάλειας της και των πολύ καλών ακολουθιών που παράγει ενδείκνυται για κρυπτογράφηση σχετικά μικρών δεδομένων, όπως για παράδειγμα σε ασύμμετρους κρυπτογραφικούς αλγόριθμους στη διαδικασία ανταλλαγής κλειδιών. Καθώς τα κλειδιά δεν επιτρέπεται να διαρρεύσουν και το μέγεθος τους δεν είναι μεγάλο (όπως θα ήταν πχ το μέγεθος ολόκληρου του μηνύματος).

4.6 Συνολικά αποτελέσματα

Οι ΓΨΑ που βασίζονται σε πρωταρχικούς LFSRs είναι γρήγοροι, εξάγουν ακολουθίες με αρκετά καλές ιδιότητες, μπορούν να κατασκευαστούν έτσι ώστε να ανταποκρίνονται στις απαιτήσεις του εκάστοτε προβλήματος και είναι εύκολα υλοποιήσιμες. Οι δυο διαφορετικές διαδικασίες εξόδου που παρουσιάστηκαν στην παρούσα είδαμε ότι δεν επιφέρουν ουσιαστικές διαφορές ως προς τους στατιστικούς ελέγχους. Ο δεύτερος, όμως, μπορεί να προσδώσει επιπλέον δυσκολίες σε έναν κρυπταναλυτή. Έχουν όμως σημαντικές παθογένειες η κύρια πηγή των οποίων είναι η γραμμικότητα τους, σε συνδυασμό με την ύπαρξη του αλγορίθμου Berlecamp-Massey η χρήση τους σε μια κρυπτογραφική εφαρμογή ως το μοναδικό μέσο κρυπτογράφηση δεν συνιστάται. Ενθαρρύνεται όμως η χρήση τους ως δομικά συστατικά ενός ευρύτερου μηχανισμού ο οποίος εμπεριέχει μη γραμμικότητα, αυτός ο μηχανισμός χρήζει μεγάλης προσοχής κατά την κατασκευή και τον έλεγχο του. Αφού όπως αναφέρθηκε με την ΓΨΑ Geffe μπορεί να βελτιώνονται αρκετοί παράμετροι, αλλά κάποιες παθογένειες να παραμένουν και η κατασκευή του μηχανισμού να δημιουργεί και άλλες (όπως η συσχέτιση της εξόδου του μηχανισμού με μια από τις εξόδους των LFSRs που χρησιμοποιούνται από αυτόν). Οι LCGs γεννήτριες είναι από τις πρώτες γεννήτριες που δημιουργήθηκαν, παράγουν πολύ καλές ακολουθίες, είναι γρήγορες, εύκολα υλοποιήσιμες, η γραμμικότητα τους όμως δεν μας επιτρέπει να τις χρησιμοποιούμε στην Κρυπτογραφία. Λόγω των ιδιοτήτων τους και της ομοιόμορφης κατανομής που παρουσιάζουν σε διαστήματα αριθμών, ενδείκνυται η χρήση τους σε άλλες εφαρμογές που δεν χρήζουν ασφαλείας, αλλά καλών στατιστικών ιδιοτήτων, όπως για παράδειγμα σε προβλήματα προσομείωσης. Οι BBS γεννήτριες παρουσιάζουν μεγάλη ασφάλεια, δεν είναι πολύπλοκες, εξάγουν ακολουθίες που είναι πολύ κοντά στις τυχαίες, είναι όμως πολύ αργές. Ενδείκνυνται για χρήση σε κρυπτογραφικές εφαρμογές, αλλά η ταχύτητα περιορίζει το πεδίο χρήσης τους σε εφαρμογές που δεν κρυπτογραφούν μεγάλο όγκο δεδομένων (όπως η ανταλλαγή κλειδιών), ή σε χρήσεις που δεν απαιτούν ταχύτητα. Η γεννήτρια SHA-1, προέκυψε μέσα από την συνάρτηση κατακερματισμού SHA-1, η οποία χρησιμοποιείται στις ψηφιακές υπογραφές στις οποίες λειτουργεί σε συνδυασμό με τον κρυπτογραφικό αλγόριθμο DSA (Digital Signature

Algorithm). Η SHA-1 ως συνάρτηση κατακερματισμού δέχθηκε μια μεγάλη επίθεση, όπως ανακοινώθηκε στις 23/2/2017 από τους CWI και Google, που αμφισβήτησε την έως τότε αψεγάδιαστη ασφάλεια του. Η ΓΨΑ SHA-1 όμως είναι από τις πιο ασφαλείς γεννήτριες έως και σήμερα. Είναι μια αποδεδειγμένα πολύ καλή γεννήτρια, που δημιουργεί ακολουθίες οι οποίες ανταποκρίνονται στους στατιστικούς ελέγχους μας. Ενδείκνυται για χρήση σε μεγάλο εύρος κρυπτογραφικών εφαρμογών. Δεν προχωρήσαμε σε αξιολόγηση της στην παρούσα με βάση την σουίτα στατιστικών ελέγχω της NIST, λόγω του ότι είναι το βασικό παράδειγμα των στατιστικών ελέγχων της σουίτας στο εγχειρίδιο χρήσης[7] της, στα πλαίσια του οποίου αξιολογείτε επαρκώς.

[7] A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications, SP 800-22, Rev1a

Παράρτημα Α΄

Αλγοριθμος Berlekamp-Massey[8]

Ο αλγόριθμος ο οποίος εκτελείται για την δυαδική ακολουθία $s_0,s_1,...,s_n$ χρησιμοποιεί τις παρακάτω παραμέτρους:

- L – τρέχουσα τιμή της πολυπλοκότητας,
- $C(D) = c_L \cdot D^L + c^{L-1} \cdot D^{L-1} + ... + c_1 \cdot D + 1$ – πολυώνυμο, η δομή της ανάδρασης για το τρέχων καταχωρητή,
- μήκος L,
- B(D)- βοηθητικό πολυώνυμο,
- N – το νούμερο του δυαδικού ψηφίου που αναλύουμε στην δοσμένη δυαδική ακολουθία.
- Η πράξη που χρησιμοποιείται είναι η XOR.
- Ο διαδοχικός αλγόριθμος αποτελείται από την ακολουθιακή εκτέλεση των παρακάτω βημάτων:

1. Ορισμός των αρχικών τιμών: C(D) = B(D) = 1, x=1, L=0, N=0.
2. Υπολογισμός για το τρέχων N bit της ακολουθίας το bit d της πρόβλεψης με το bit του παραγόμενου τρέχων πολυώνυμου C(D) με L προυπάρχοντα bits της ακολουθίας: $d = s_N + \sum_{j=1}^{L} c_j \cdot s_{N-j}$
3. Εάν d=0, τότε το bit s_N ορθά παράγει τον καταχωρητή μήκους L με δομή ανάδρασης, με δοσμένο C(D) , x:=x+1, μεταφορά στο βήμα 6.
4. Εαν d=1 και $2 \cdot L > N$, τότε το bit s_N εσφαλμένα παράχθηκε στο καταχωρητή ανάδρασης, για το δοσμένο πολυώνυμο C(D), αλλά η ορθή παραγωγή του bit s_N μπορεί να γίνει με την διόρθωση του πολυώνυμου C(D) χωρίς την αύξηση του μήκους L του καταχωρητή σύμφωνα με την έκφραση: $C(D):=C(D)+D^x \cdot B(D)$, x:=x+1.
5. Ένα d =1, αλλά $2 \cdot L \leq N$, τότε εσφαλμένα έγινε η παραγωγή s_N του τρέχοντος καταχωρητή μήκους L και η ανάδραση, της δοσμένης C(D) μπορεί να πραγματοποιηθεί με την αύξηση του μήκους του καταχωρητή L:=N+1-L και με διόρθωση του πολυώνυμου $C(D):=C(D)+D^x \cdot B(D)$, για αυτό στο B(D) αποθηκεύεται η προηγούμενη τιμή του πολυώνυμου C(D), και η τιμή x ορίζεται ως μονάδα x:=1.
6. Εκτέλεση μεταφοράς στο επόμενο σύμβολο της ακολουθίας: N:=N+1, εάν N=n , τότε πάμε στο τέλος, αλλιώς επιστροφή στο βήμα 2 του αλγόριθμου.

Παράδειγμα[1]

[8] Ν. Μπάρδης Κρυπτογραφία και Συστήματα Ασφαλείας Πληροφοριών

Έστω ότι δίδεται η ακολουθία:

S= $s_0, s_1,..., s_{14}$=1 0 1 0 1 1 0 0 1 0 0 0 1 1 1

Η αρχική κατάσταση είναι: C(D)=1; B(D)=1; x=1, N=0, L=0.

N=0. Καθώς =0, τότε το bit δεν μπορεί να προβλεφτεί και d = x_0 = 1.

Καθώς 2·L=N, τότε πάμε στο βήμα 5, C(D)=C(D)+D^1·B(D)=1+D,

x:=1, B(D)=1; L=0+1-0=1.

- N=1 Καθώς C(D)=1+D^1, τότε το bit που θέλουμε να προβλέψουμε επαναλαμβάνεται όπως το προηγούμενο: d = x_1+x_0·1 =1. Η διόρθωση του πολυώνυμου εκτελείται στο βήμα 4, καθώς εκτελείται 2·L>N C(D)=C(D)+D^1·B(D)=1+D+D=1, x:=2.
- N=2 καθώς στον τύπο για C(D) δεν υπάρχει κανένα D , τότε το bit που είναι να προγνωστεί ισούται 0: d = x_2+0=1+0=1. Καθώς 2·L=N τότε πάμε στο βήμα 5 C(D):=C(D)+D^2·B(D) = 1+D^2. B(D)=1, x:=1; L=2+1-1=2; Αυτό σημαίνει ότι για την πρόβλεψη του επόμενου bit πρέπει να ληφθούν υπόψη τα 2 προηγούμενα bits, καθώς στο C(D) περιέχεται D^2, τότε η πρόβλεψη εκτελείται στο πρώτο bit με την ανάλυση του ζευγαριού.
- N=3 Τα προηγούμενα bits s_1 s_2 =0 1,

  το πρώτο αντιστοιχεί στο D^2 , και

  το δεύτερο – D^1,

  καθώς στο C(D) μπαίνει μόνο το D^2, τότε το bit που είναι να προβλεφτεί ισούται με το πρώτο – αυτό σημαίνει ότι s_1=0 d=s_3+0 = 0.

  Εκτελείται το βήμα 3

  x:=x+1=2.
- N=4 Τα προηγούμενα bits s_2 s_3 =1 0,

  το πρώτο από αυτά αντιστοιχεί στο D^2 , και

  το δεύτερο – D^1, καθώς στο C(D) περιέχεται μόνο το D^2, τότε το bit που είναι να προβλεφτεί ισούται με το πρώτο –

  δηλαδή s_2=1 d=s_4 + 1 = 1+1=0.

  Εκτελείται το βήμα 3

  x:=x+1=3.
- N=5 Τα προηγούμενα bits s_3 s_4 =0 1,

  το πρώτο από αυτά αντιστοιχεί στο D^2 , και

  το δεύτερο – D^1, καθώς στο C(D) περιέχεται μόνο το D^2, τότε το bit που είναι να προβλεφτεί ισούται με το πρώτο –

  δηλαδή s_3=0 d=s_5 + 0 = 1.

  2·L<N C(D):=C(D)+D^3·B(D) = 1+D^2+D^3.

  B(D) = 1+D^2,

L=5+1-2=4.

x:=1.

- N=6 Καθώς L=4 η πρόγνωση έχει υλοποιηθεί για τα 4 προηγούμενα bits s_2 s_3 s_4 s_5 = 1 0 1 1 . Καθώς C(D) περιέχει μόνο το D^2 και D^3 (s_2 αντιστοιχεί στο D^4, s_3 αντιστοιχεί στο D^3 , s_4 αντιστοιχεί στο D^2, s_5 αντιστοιχεί στο D^5), τότε η πρόγνωση εκτελείται ως το άθροισμα όλων των bits s_3+s_4= 0+1=1. $d=s_6+(s_3+s_4)$ = 0+(0+1)=1. Γίνεται μεταφορά στο βήμα 4

  $C(D):=C(D)+D^1 \cdot B(D)=1+D^2+D+D\cdot(1+D^2) = 1+D+D^2$.

  x:=2.

- N=7 Η πρόγνωση πάλι εκτελέστηκε στα 4 προηγούμενα s_3 s_4 s_5 s_6 και καθώς στο C(D) περιέχεται μόνο τα D^2 και D^1 τότε η πρόγνωση ισούται με το άθροισμα των δύο μικρότερων σε τάξη

  $s_5 + s_6 = 1+0 =1$ οπότε $d=s_7 + 1=1$.

  Καθώς $2\cdot L=8 > N$ τότε η διόρθωση του πολυώνυμου εκτελείται στο βήμα 4

  $C(D)=C(D)+D^2 \cdot B(D) = 1+D+D^2 + D^2\cdot(1+D^2) = 1+D+D^4$.

  x:=3

- N=8 Η πρόγνωση πάλι εκτελέστηκε στα 4 προηγούμενα s_4 s_5 s_6 s_7 και καθώς στο C(D) περιέχεται μόνο τα D^4 και D^1 τότε η πρόγνωση ισούται με το άθροισμα μόνο με τα δύο ακριανά από αυτά bits

  $s_4 + s_7 = 1+0 =1$ συνεπώς $d=s_8+1=0$.

- Στη συνέχεια είναι εύκολο να δούμε ότι η πρόβλεψη πάντα θα συμπίπτει με το bit της ακολουθίας και η πολυπλοκότητα δεν θα αυξάνει μέχρι το τέλος της ακολουθίας.

ΠΑΡΑΡΤΗΜΑ Β′

Στο παρόν παράρτημα παρέχονται τα μέσα που χρησιμοποιήθηκαν για την παραγωγή των ακολουθιών που εξετάστηκαν στο Κεφάλαιο 4.

Πρωταρχικος LFSR και παραλλαγή μεγεθους 20-bit[9]

```
def prtReg(L):
'''
Nicely prints out the list as a sequence of bits
'''
#  R = []
#  for bit in L:
#    if L[bit] == 0:
#      R.append("_")
#    else:
#      R.append("#")
#  for bit in R:
for bit in L:
print(bit, end="")
print()

def lShift(L):
'''
Shifts left the list's contents. The last element is filled in
according to the formula
at the end of the function.
'''
A = L[0]
B = L[len(L)-2]
C = L[len(L)-1]
for j in range(len(L)-1):
L[j] = L[j+1]
L[len(L)-1] = A ^ B ^ C

def rShift(L):
'''
Shifts right the list's contents. The last element is filled in
according to the formula
at the end of the function.
'''
A = L[len(L)-1]
B = L[len(L)-3]
for j in range(len(L)-1, 0, -1):
L[j] = L[j-1]
L[0] = A ^ B

def main():
```

[9] Χρησιμοποιείται γλώσσα προγραμματισμού Python, περισσότερες πληροφορίες μπορούν να βρεθούν στον ιστότοπο:

https://github.com/mellowiz/LFSR?fbclid=IwAR3bx4cAhtfBnifgVKQeu8cOzxotlPEdzpXybxdHw8_t6TNddEg8Lv9EfXM

```
   # L0 is the initial value of the list/register.
   L0 = [0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0,
0, 1, 0, 1, 1]
   #  L0 = [0, 1, 0, 0, 1]
   L = list(L0) # "L" is created as a new list (i.e. not a copy),
holding the same values
   print("Iteration:\t", 0, end="\t")
   print("\033[91m", end="")
   for bit in L0:
   print(bit, end="")
   print("\033[0m")
   for n in range(2**len(L)):
   rShift(L)
   print("Iteration:\t", n+1, end="\t")
   prtReg(L)
   if L == L0:
   break

   if __name__ == '__main__':
   main()
```

Geffe[10]

```
public
class
GeffeGenerator
{
                    private LFSR lfsr1;
                    private LFSR lfsr2;
                    private LFSR lfsr3;
                    private StringBuffer gamma;
                    public GeffeGenerator(LFSR lfsr1, LFSR lfsr2, LFSR
                lfsr3) {
                    this.lfsr1 = lfsr1;
                    this.lfsr2 = lfsr2;
                    this.lfsr3 = lfsr3;
                    this.gamma = new StringBuffer();
                    }
                    public int step() {
                    int x = lfsr1.shift();
                    int y = lfsr2.shift();
                    int s = lfsr3.shift();
                    int out = (s & x) ^ ((1 ^ s) & y);
                    gamma.append(out);
                    return out;
                    }
                    public void step(int n) {
                    for (int i = 0; i < n; i++){
                    step();
                    }
                    }
                    public String getGamma() {
                    return gamma.toString();
                    }
                    }
```

[10] Χρησιμοποιείται γλώσσα προγραμματισμού Java, περισσότερες πληροφορίες μπορούν να βρεθούν στον ιστότοπο: https://github.com/NikitaDoroshkin/geffe-generator

```
    public
class
Polynomial
{
                private long value;
                private int degree;
                Polynomial(int ... coefficients) {
                if (coefficients.length == 1) {
                this.value = coefficients[0];
                } else {
                this.value = 1;
                for (int i : coefficients) {
                this.value ^= ((long)1 << i);
                }
                }
                setDegreeOfValue();
                }
                public int getDegree(){
                return degree;
                }
                private void setDegreeOfValue() {
                this.degree =
            getDegreeOfLongValue(this.value);
                }
                public static int getDegreeOfLongValue(long
            val) {
                return 63 - Long.numberOfLeadingZeros(val);
                }
                public long getValue() {
                return value;
                }
                }
```

```
    import
java.io.BufferedReader;
                                import java.io.File;
                                import java.io.FileReader;
                                import java.io.IOException;
                                public class Main {
                                public static void main(String[] args) throws
                            IOException {
                                File file = new File("src/result.txt");
                                BufferedReader br = new BufferedReader(new
                            FileReader(file));
                                String outputSequence = br.readLine();
                                Polynomial polynomial1 = new Polynomial(30, 6, 4, 1);
                                Polynomial polynomial2 = new Polynomial(31, 3);
                                Polynomial polynomial3 = new Polynomial(32, 7, 5, 3,
                            2, 1);
                                LFSR lfsr1 = new LFSR(polynomial1);
                                LFSR lfsr2 = new LFSR(polynomial2);
                                LFSR lfsr3 = new LFSR(polynomial3);
                                lfsr1.setInitialState(651497879);
                                lfsr2.setInitialState(1259760270);
                                lfsr3.setInitialState(2229332000L);
                                GeffeGenerator geffe = new GeffeGenerator(lfsr1,
                            lfsr2, lfsr3);
                                geffe.step(outputSequence.length());
                                System.out.println("LFSR1 result sequence:\n" +
                            lfsr1.getOutputSequence());
                                System.out.println("LFSR2 result sequence:\n" +
                            lfsr2.getOutputSequence());
                                System.out.println("LFSR3 result sequence:\n" +
                            lfsr3.getOutputSequence());
                                System.out.println("Geffe gamma sequences (calculated
                            and expected)\n" + geffe.getGamma() + "\n" +
                            outputSequence);
                                }
                                }
```

```
    import
java.util.HashMap;
                        import java.util.Map;
                        public class LFSR {
                        private long polynomial;
                        private int polynomialDegree;
                        private long initialState;
                        private long currentState;
                        private StringBuffer outputSequence;
                        public LFSR(Polynomial initPolynomial, long initialState) {
                        this.polynomialDegree = initPolynomial.getDegree();
                        this.polynomial = initPolynomial.getValue() ^ ((long)1 <<
                    polynomialDegree);
                        setInitialState(initialState);
                        }
                        public LFSR(Polynomial initPolynomial) {
                        this(initPolynomial, 1);
                        }
                        public int shift() {
                        int out = (int) (currentState & 1);
                        int tempBit = (Long.bitCount((currentState & polynomial)))
                    & 1;
                        currentState >>= 1;
                        currentState ^= ((long)tempBit << (polynomialDegree - 1));
                        outputSequence.append(out);
                        return out;
                        }
                        public void setInitialState(long initialState) {
                        this.initialState = initialState;
                        this.currentState = this.initialState;
                        refreshOutputSequence();
                        }
                        private void refreshOutputSequence() {
                        this.outputSequence = new StringBuffer();
                        }
                        public String getOutputSequence() {
                        return outputSequence.toString();
                        }
                        }
```

LCG και BBS

Οι γεννήτριες LCG και BBS παρέχονται ενσωματωμένες στη σουίτα στατιστικών ελέγχων της NIST (A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications).

Βιβλιογραφία

[1] A.Barg, ENEE 626: *Error Correcting Codes*, University of Meryland lectures (2015)

[2] A.J.Menezes, P.C. van Oorschot, S.A. Vanstone , *Handbook of Applied Cryptography*, CRC (1997)

[3] A.Rukhin, J.Soto, J.Nechvatal, M.Smid, E.Barker, S.Leigh, M.Levenson, M.Vangel, D.Banks, A.Heckert, J.Dray, S.Vo, *A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications*, SP 800-22, Rev1a, NIST (2010)

[4] B.Schneier , *Applied Cryptography Protocols*, Algorithms and Source Code in C,Wiley (1995)

[5] H.Niederreiter, K.H.Robinson, *Complete Mappings of Finite Fields*, Australian Mathematical Society (1982)

[6] John B. Fraleigh, Victor J. Katz, *A first course in abstract algebra*, Addison-Wesley (2003)

[7] Mullen G.L., Panario D., *Handbook of Finite Fields*, CRC (2013)

[8] NIST, *Digital Signature Standard* (DSS), FIPS PUB 186-2 (2000)

[9] NIST, *Secure Hash Standard*, FIPS PUB 180-1 (1993)

[10] Rudolf Lidl, Harald Niederreiter, *Encyclopedia of Mathematics and its Applications*, Finite Fields-Cambridge University Press (1997)

[11] Rudolf Lidl, Harald Niederreiter, *Introduction to Finite Fields and their Applications*, Cambridge University Press (1986)

[12] Solomon W. Golomb, *Shift Register Sequences*, Holden-Day, Inc. (1967)

[13] V.Shoup, *A computational introduction to number theory and algebra*, Cambridge University Press (2009)

[14] Α. Βασιλειάδης, Ν.Παπανικόλας, Λ.Λεωνιδόπουλος, *Αξιολόγηση Γεννητριών Παραγωγής Ψευδοτυχαίων Ακολουθιών*, Αθήνα 2016

[15] Α.Κοντογεώργης, *Πεπερασμένα Σώματα και Κρυπτογραφία*, Πανεπιστημιακές Σημειώσεις Αθήνα (2012)

[16] Α.Ν.Βενέτη, *Η θραυσματική διάσταση ως μέτρο αξιολόγησης γεννητριών ψευδοτυχαίων αριθμών*, Πάτρα (2014)

[17] Α.Παγουρτζής, Π.Πόντικας, *Κρυπτογραφία Ψευδοτυχαιότητα-Κρυπτοσυστήματα ροής*, ΣΗΜΜΥ ΕΜΠ

[18] Β.Κάτος, Γ.Στεφανίδης, *Τεχνικές Κρυπτογραφίας και Κρυπτανάλυσης*, Ζυγός

(2003)

[19] Γ. Αντωνιάδης, Α.Κοντογεώργης, *Πεπερασμένα Σώματα και Κρυπτογραφία*, Ελληνικά Ακαδημαϊκά Ηλεκτρονικά Συγγράμματα και Βοηθήματα, Κάλλιπος (2015)

[20] Γ.Χ.Σιδηρόπουλος, *Σύγχρονες Ηλεκτρονικές Τεχνολογίες*, Ιωάννινα (2008)

[21] Γ.Χ.Στεφανίδης, *Κρυπτογραφία, Πανεπιστημιακές Σημειώσεις*, Παν.Μακ.

[22] Δ. Βάρσος, Δ. Δεριζιώτης, Ι. Εμμανουήλ, Μ.Μαλιάκας, Ο.Ταλέλλη, *Μια εισαγωγή στην Άλγεβρα, εκδόσεις Σοφία*, Γ΄έκδοση (2012)

[23] Δ.Μ. Πουλάκης, *Κρυπτογραφία η επιστήμη της ασφαλούς επικοινωνίας*, εκδόσεις Ζήτη, Θεσσαλονίκη (2006)

[24] Ε. Ρέζου, *Υπολογισμός συνάρτησης μεταφοράς μέσω συναρτήσεων συσχέτισης*, Πάτρα (2014)

[25] Ε. Σαλούστρος, *Μελέτη σχεδίαση και υλοποίηση σε FPGA προγραμματιζομένων LFSR με δυνατότητα παραγωγής τόσο της ορθής όσο και της αντίστροφης ακολουθίας τους*, Χανιά (2012)

[26] Ε.Ζάχος, Α.Παγουρτζής, Π.Γρόντας, *Υπολογιστική Κρυπτογραφία*, Ελληνικά Ακαδημαϊκά Ηλεκτρονικά Συγγράμματα και Βοηθήματα, Κάλλιπος (2015)

[27] Θ.Αποστολίδη, Χ.Χαραλάμπους, *Θεωρία Galois*, , Ελληνικά Ακαδημαϊκά Ηλεκτρονικά Συγγράμματα και Βοηθήματα, Κάλλιπος (2015)

[28] Κ.Στάικος, *Σχεδίαση Γεννητριών Τυχαίων Αριθμών Χαμηλής Κατανάλωσης Ισχύος*, Πάτρα (2009)

[29] Κ. Λημνιώτης, *Κρυπτογραφία, Πανεπιστημιακές Σημειώσεις*, Αθήνα (2008)

[30] Ν.Ε.Κολοκοτρώνης, *Μη-γραμμική Επεξεργασία Σήματος και Εφαρμογές στην Κρυπτογραφία*, Αθήνα (2003)

[31] Ν.Μπάρδης, *Κρυπτογραφία και Συστήματα Ασφαλείας Πληροφοριών*, Πανεπιστημιακές Σημειώσεις Αθήνα (2014)