გ.კოტრიკაძე. კრიპტოგრაფიული მეთოდების მიმოხილვა, 2014წ. (ბაკალავრიატი 2)

კრიფტოგრაფიული მეთოდების მოკლე მიმოხილვა

კრიპტოგრაფიას დიდი ხნის ისტორია აქვს. მას ჯერ კიდევ ცეზარის დროს იყენებდნენ, თუმცა მისი თეორიული საფუძვლები მხოლოდ 1948 წელს ჩამოყალიბა კლოდ შენონმა [3]. ინფორმაციის კრიპტოგრაფიული დაცვა ძირითადად გამოიყენებოდა სახელმწიფო სტრუქტურებში, სამხედრო საქმესა და დიპლომატიაში. ინფორმატიზაციის მზარდმა განვითარებამ მოითხოვა საბირჟო, საბანკო, კომერციული ინფორმაციის, იურიდიული დოკუმენტების, ავადმყოფობის ისტორიებისა და ბევრი კრიპტოგრაფიის სამთავრობო სტანდარტები.

რისთვის არის საჭირო ინფორმაციის დაცვა? რისი გაკეთება შეუძლია ინფორმაციის გამტაცებელს - ჰაკერს? მას შეუძლია შეცვალოს ინფორმაცია თავისი მიზნებისთვის, გაიფართოვოს თავისი კანონიერი უფლებამოსილებანი, გაიგოს, ვის რა ინფორმაციასთან აქვს შეხება, შეუშალოს ხელი მომხმარებლებს შორს ინფორმაციის გაცვლას. კრიპტოგრაფია იცავს ინფორმაციას, როგორც არასანქცირებული შეღწევებისაგან, ისე კომპიუტერული ვირუსისაგან.

ტერმინი „კრიფტოგრაფია“ მოიცავს სამეცნიერო-ტექნიკურ სფეროს, რომელიც იყოფა ორ ძირითად ნაწილად:

კრიფტოგრაფიული სისტემის სინთეზი და კრიფტოანალიზი. პირველი ცდილობს, შექმნას ინფორმაციის დაშიფრა-დასაიდუმლოების მეთოდები, ხოლო კრიფტოანალიზი ცდილობს, გამოიკვლიოს ან „გატეხოს“ მეთოდები.

დაშიფვრა გულისხმობს ორი პროცედურის რეალიზაციას: ინფორმაციის საწყისი ღია ტექსტის დაშიფრა, დაშიფრული ტექსტის მიღების მიზნით; დაშიფრული ტექსტიდან საწყისი ტექსტის აღდგენა- დეშიფრაცია.

ორივე შემთხვევაში ადგილი აქვს ტექსტის გარდაქმნას განსაზღვრული ალგორითმის სიმრავლიდან, რომელიც ქმნის კრიპტოგრაფიული სისტემას: სისტემის ნაწილს, რომელიც ახორციელებს ინფორმაციული ტექსტის კონკრეტულ გარდაქნას, ეწოდება გასაღები. როგორც წესი (თუმცა, არა ყოველთვის), გასაღების სიგრძე გაცილებით ნაკლებია ტექსის სიგრძეზე.

თანამედროვე კრიპტოალგორითმები იყოფა ორ კლასად - სიმეტრიული და ასიმერტიული. შიფრაციის სიმეტრიულ სისტემებში[3] დასაშიფვრის გასაები (საიდუმლო) ძირითადად ემთხვევა დეშიფრაციის საიდუმლო გასაღებს, ხოლო შიფრაციის ასიმეტრიულ სქემებში (კრიპტოგრაფია ღია გასაღებით) დაშიფვრის ღია გასაღები არ ემთხვევა დეშიფრაციის საიდუმლო გასაღებს (აღსანიშნავია, რომ ინფორმაციის დაცვა, საზოგადოდ,

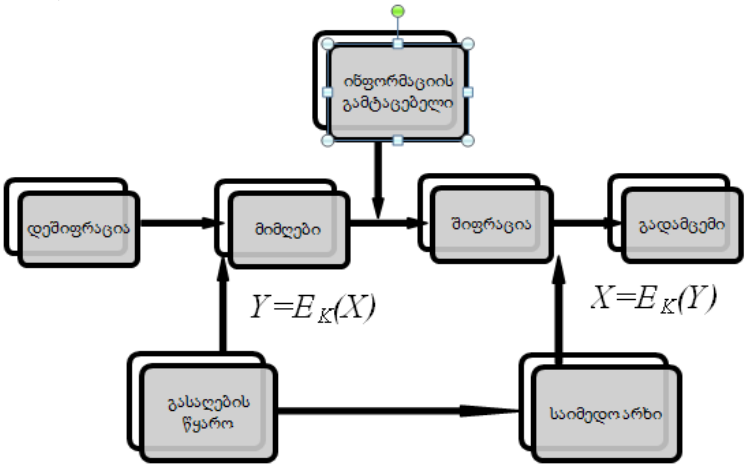
კომპიუტერულ სისტემებში გაცილებით ფართოა ცნებაა, რასაც მხოლოდ [2] მონოგრაფიაც კი საკმაოდ ნათლად წარმოაჩენს).

კრიპტოგრაფიაში მიღებულია კერკჰოფის წესი: შიფრის მედეგობა უნდა განისაზღვრებოდეს მხოლოდ გასაღების საიდუმლოობით. ინფორმაციის გამტაცებელს ან კრიპტოანალიტიკოსს შეუძლია, იცოდეს ყველა მონაცემი, გარდა გასაღებისა. ითვლება, რომ კრიპტოსისტემა გახსნილია, თუ გამტაცებელს დასაშვებზე მეტი ალბათობით შეუძლია შემდეგი ოპერაციების ჩატარება; საიდუმლო გასაღების პოვნა, გარდაქმნის ეფექტური ალგორითმის შესრულება, რომელიც ფუნქციონალურად ექვივალენტურია საწყისი კრიპტოალგორითმისა.

იმისთვის, რომ კრიფტოსისტემა გახსნილად ჩაითვალოს, საჭიროა არა მხოლოდ გასაღების გახსნის (ანუ საიდუმლო გასაღების პარამეტრების მიღების) ალგორითმის ჩვენება, არამედ იმის ჩვენებაც, რომ ეს ალგორითმი შეიძლება შესრულდეს რეალურ დროში. ალგორითმის სირთულე ითვლება კრიპტოსქემის ერთ-ერთ მთავარ მახასიათებლად და მას კრიპტომედეგობა ეწოდება.

§1. მეთოდები, რომლებიც ითვალისწინებს გასაღების საიმედო არხით გაცვლას

სიმეტრიული (შენონისეული) კრიპტოსისტემის კლასიკური მოდელი შეიძლება შემდეგნაირად წარმოვიდგინოთ:



სურ:1.1

ამ მოდელში სამი მონაწილეა: გადამცემი, მიმღები და ინფორმაციის გამტაცებელი. გადამცემის ამოცანაა, ღია არხით გადასცეს შეტყობინება დაცული ფორმით. ამისთვის ის საწყის X ტექსტს დაშიფრავს K გასაღებით და გასცემს დაშიფრულ Y ტექსტს. მიმღების ამოცანაა, გაშიფროს მიღებული Y ტექსტი და აღადგინოს X შეტყობინება. იგულისხმება, რომ გადამცემს აქვს გასაღბის საკუთარი წყარო და მის მიერ გენერირებულ გასაღებებს წინასწარ საიმედო არხოთ (სპეციალური კურიერი) გადასცემს მიმღებს. ინფორმაციის გამტაცებლის (ჰაკერის) ამოცანაა, წაიკითხოს გადაცემული შეტყობინებები.

არსებობს კრიპტოსისტემის მრავალი მეთოდი. მაგალითად, ცეზარის, ვიჟინერის, ვერნამის და სხვა, რომლებშიც გამოყენებულია x ინფორმაციის გარდასახვის სხვადასხვა ფუნქცია *y=f(x),* რომელთათვისაც არსებობს შებრუნებული f(y)=x ფუნქციები, რაც შეიძლება იყოს სიმბოლოთა ჩვეულებრივი გადანაცვლება, ჩასმა, მოდულით შეკრება-გამრავლების ოპერაციები და ა.შ [3].

ცეზარის მეთოდი: ანბანის ყოველ სიმბოლოს შეესაბამება გარკვეული რიცხვი, მაგალითად, ანბანში მისი ადგილის შესაბანისი ნომერი. ტექსტის ყოველ სიმბოლოს ემატება ფიქსირებული სიმბოლო (იკრიბება მათი შესაბამისი რიცხვები ფიქსირებული მოდულით, რომელიც ტოლია ანბანში სიმბოლოების რაოდენობისა) და იწერება მიღებული რიცხვის შესაბამისი სიმბოლო. მაგ:

ინფორმაცია: A B C D E

გასაღბი: D D D D D

კრიპტოგრამა: E F G H I

ვიჟინერის მეთოდი: მთელი ტექსტი იშიფრება მისი სიგრძის ტოლი შფრით. შიფრაცია ხდება ანალოგიურად. ეს არის გაუხსნელი მეთოდი, თუ ერთი შიფრი გამოიყენება მხოლოდ ერთხელ.

ამ მეთოდზე და ასევე, ჩანაცვლებებსა და გადანაცვლებებზე არის აღებული კლასიკური კრიპტოსისტემის

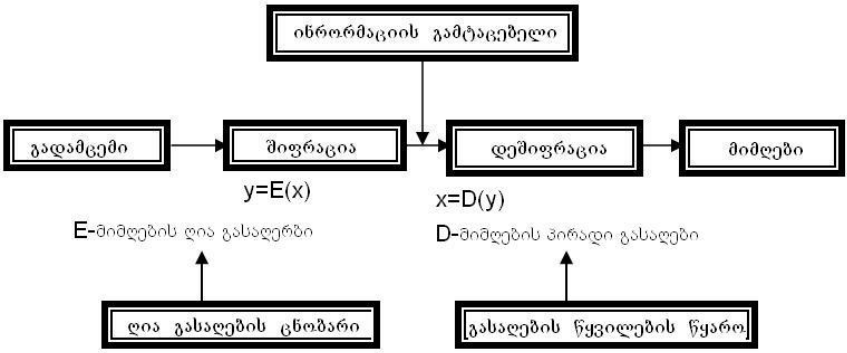
სტანდარტი DES (Data Encryption Standart) [13]. იგი აგებულია ბლოკური კრიფტოალგორითმით (ბლოკის სიგრძეა 64 ბიტი) 64 ბიტის სიგრძის გასაღებით, რომლის პოზიციათაგან 56 შემთხვევითია, ხოლო 8 გამოიყენება გასაღბის კონტროლოსთვის. ეს ალგორითმი დამუშავებულია ფირმა IBM-ის მიერ და რეკომენდირებულია სტანდარტების ნაციონალური ბიუროს მიერ ეკონომიკის ღია სექტირებში გამოყენებისათვის. DES არის ძალიან ადვილად რეალიზებადი და სწრაფქმედებადი, მაგრამ მისი უმთავრესი ნაკლი არის ის, რომ გასაღების პერიოდულად შესაცვლელად საჭიროა საიდუმლო არხი (სპეციფიკური). როცა ქსელში n მომხმარებელია, საჭიროა $\frac{n(n-1)}{2}$ გასაღების გენერაცია და გაცვლა (რადგან ყოველ წყვილს თავისი შიფრი უნდა ჰქონდეს), რაც ძალიან რთულდება დიდი n-ის დროს.

§2. გასაღბის ღია არხით გაცვლის მეთოდები

არსებობს გასაღების სისტემის მართვის ამოცანის გადაჭრის და ღია არხით (საიდუმლო არხის გარეშე) კრიპტოგრაფიული კავშირის სხვადასხვა მეთოდი. ერთ-ერთი მათგანი დამყარებულია საიდუმლო გასაღების ღია გაცვლის იდეაზე. მისი არსის გასაგებად გამოვიყენოთ შემდეგი ანალოგია: ვთქვათ, არსებობს ორი ლექსიკონი: X-Y (მაგალითად ქართულ-ინგლისური), რომელიც ყველასთვის მისაწვდომია და Y-X (ინგლისურ-ქართული), რომელიც აქვს მხოლოდ ერთ

პიროვნებას. ქართული ტექსტის ინგლსურად თარგმნა შეუძლია ყველას, ხოლო ინგლისური ტექსტის ქართულად, საჭიროა ქართულ-ინგლისური ლექსიკონის მთლიანი გადარჩვა, რაც ძალიან დიდ დროს მოითხოვს და პრაქტიკულად შეუძლებელია. ასევეა კრიპტოგრაფიაშიც. ყოველი მომხმარებელი ღაი არხში აცხადებს თვის X-Y ლექსიკონს, ხოლო Y-X ლექსიკონი აქვს მხოლოდ მას. ყველას შუძლია, გაუგზავნოს მას ინფორმაცია, ხოლო წაკითხვა მხოლოდ მას შეუძლია.

სწორედ ეს თვისებები უდევს საფუძვლად გასაღბების ღია განაწილების მეთოდებს, რომელთაც ასიმეტრიულ მეთოდებსა აწოდებენ. ამ მეთოდებშიც მაღალ კრიპტომედეგობას განაპირობებს გამოთვლების დიდი სირთულე. ღია გასაღებების მოდელი შეგვიძლია შემდეგნაირად წარმოვიდგინოთ:



სურ.1.2.

სისტემის მოდელში სამი მონაწილეა: გადამცემი, მიმღები და ინფორმაციის გამტარებელი. გადამცემი X ტექსტს

დაშიფრავს მიმღბის ღია E გასაღბით და გაგზავნის დაშიფრულ Y ტექსტს. მხოლოს მიმღებს შეუძლია Y ტექსტის გაშიფვრა და X-ის წაკითხვა, რადგან მხოლოდ მას აქვს საიდუმლოი D გასაღები. იგულისხმება, რომ გამგზავნს აქვს გასაღებების წინასწარ ან საიმედო არხით გადაცემემა გადამცემს. ასე რომ, N-აბონენტიან ქსელში ყოველი აბონენტი გამოიმუშავებს გასაღბების საკუთარ წყვილს (E,D) და აქვეყნებს E--ს. ე.ი. ქსელში იქნება შირაციის N ღია გასაღბები და დეშიფრაციის N საიდუმლო გასაღები. ეს ხსნის $\frac{N(N-1)}{2}$ გასაღბის საჭიროების პრობლემას.

1. დიფი-ჰელმან მერკლეს მეთოდი [14].

ალგორითმი ეყრდნობა გალუას GF(p) მარტივ ველში ლოგარითმების გამოთვლის სირთულეს. ვთქვათ, $y = a^x mod\, p \quad (1 \le x \le q-1)$, სადაც a ველის პრიმიტიული ელემენტია. ამბობენ, რომ x არის y-ის ალგორითმი a ფუძით GF(p) ველში: $x = log_a Y \quad (1 \le y \le p-1)$ y-ის გამოთვლა x-ის მეშვეობით არ წარმოადგენს სირთულეს და მოითხოვს მაქსიმუმ $2 \cdot log_2 p$ - გამრავლების ოპერაციას, მაგალითად: $a^{18} = (((a^2)^2)^2)^2 \cdot a^2$ [15]. მეორე მხრივ, x-ის გამოთვლა y-ის მეშვეობით გაცილებით უფრო რთულია და მოცემული p-სთვის

ცნობილი საუკეთესო ალგორითმების გამოყენებითაც კი $p^{1/2}$ ოპერაციათა რაოდენობას მოითხოვს [15].

გასაღბის გაცვლა ხორციელდება სქემით: ცნობილია ორი მარტივი: p (მარტივი) და a (მთელი). $(1 \leq X_i, X_j \leq p-1)$.

i-ური მომხმარებელი გამოთვლის $y_{i=}a^{x_i} mod\ p$ რიცხვს და ღია ფაილით უგზავნის მას j-ურ მომხმარებელს. თავის მხრივ, j-ური მომხმარებელი გამოითვლის $y_j = a^{x_j} mod\ p$ რიცხვს და ღიად უგზავნის მას i-ურ მომხმარებელს.

i-ური მომხმარებელი y_j-ის მეშვეობით აფორმებს გასაღებს: $g_{ij} = y_j^{x_i} modp$

j-ური მომხმარებელი y_i-ის მეშვეობით აფორმებს გასაღებს: $g_{ji} = y_i^{x_j} modp$

ეს გასაღებები იდენტურია, რადგან:

$$y_j^{x_i} = (a^{x_j})^{x_i} = a^{x_j x_i} = (a^{x_i})^{x_j} = y_i^{x_j} (modp)$$

ე.ი. $g_{ij} = g_{ji}$.

2. რაივესტ, შამირისა და ეიდელმანის ალგპრითმი- RSA.

განსხვავებით I ალგორითმისგან, RSA ახორციელებს დაშიფრული ინფორმაციის ღის არხით გადაცემას. ალგორითმი ეყრდნობა ეილერის ცნობილ თეორემას: $x^{\varphi(N)} \equiv x mod N$, სადაც φ(N) ეილერის ფუნქციაა - N - ზე ნაკლები და მასთან ურთიერთ მარტივი რიცხვების რაოდენობა [17],$(1 \leq x \leq N)$.

საზოგადოდ, φ(N)-ის გამოთვლა დიდი რიცხვებისთვის

რთულია, მაგრამ ცნობილია, რომ $\varphi(N) = (p-1)(q-1)$, როდესაც N=pq, სადაც p და q ნებისმიერი მსრტივი რიცხვებია.

ალგორითმის განხორციელებისას მოიძებნება მაღალი რიგის ორი მარტივი რიცხვი p და q, რომლებიც საიდუმლოდ ინახება, ხოლო N გამოცხადდება (ყველასთვის, Y და Z-ისთვის), საიდუმლოდ რჩება გამოთვლილი φ(N). შემდეგ მოიძებნება ისეთი e და d რიცხვები, რომ $ed = 1 mod \varphi(N)$, (სადაც e ურთიერთმარტივია φ(N) -თან) [15]. ამის შემდეგ გამოცხადდება.

ინფორმაცია გადაეცემა ისეთი $M_1, M_2 \ldots$ მთელი რიცხვების მიმდევრობით, რომელთაგან თითოეული M მოთავსებულია [0, N-1] ინტერვალში. M ინფორმაციის დაშიფრა ხდება შემდეგნაირად: $C = M^e mod\ N$, სადაც C არის დაშიფრული ტექსტი. $ed = 1 mod \varphi(N)$ ანუ $ed = k \cdot \varphi(N) + 1$, რადგან $X^{k \cdot \varphi(N)+1} = X mod\ N$.

ნებისმიერი მთელი X-ისთვის [0,N-1] შუალედიდან და ნებისმიერი k-სთვის დეშიფრაცია შეიძლება განხორციელდეს C-ს აყვანით d-ხარისხში: $C = M^{ed} = M^{k \cdot \varphi(N)+1} = M mod\ N$.

მიმღები აცხადებს e და N რიცხვებს, რომელთა მეშვეობითაც გადამცემი დაშიფრავს ტექსტს, ხოლო გაშიფვრა შეუძლია მხოლოდ მიმღებს, რადგან მხოლოს მას აქვს საიდუმლო გასაღები- d რიცხვი. ჰაკერისთვის ძირითადი სირთულე მდგომარეობს φ(N)-ის გამოთვლაში, რადგან მისთვის უცნობია p და q რიცხვები, ხოლო დიდი მნიშვნელობის N-

ისთვის φ(N)-ის გამოთვლა ძნელია. ალგორითმის მედეგობა ემყარება მთელი რიცხვების მამრავლებად დაშლის სირთულეს.

კრიპტოგრაფიული სისტემა შეგვიძლია შემდეგნაირადაც წარმოვიდგინოთ: არის ორი მომხმარებელი: X და Y. X ირჩევს N_1, e_1 და d_1 რიცხვებს, ხოლო Y-N_2, e_2 და d_2-ს. e_1, N_1 და e_2, N_2 გამოცხადებულია (ცნობილია ყველასთვის). როდესაც X უგზავნის Y-ს ინფორმაციას, გოგო ტექსტს დაშიფტავს ჯერ საკითსტი d_1 და N_1-ით, ხოლო შემდეგ- Y-ის e_2 და N_2-ით:

$$C_1 - M^{d_1} mod N_1, \quad C - C_q^{e_2} mod N_2 = (M^{d_1} mod N_1)^{e_2} mod N_2.$$

ტექსტის გასაშიფრად Y ჯერ გამოიყენებს საკუტარ d_2-ს და აღადგენს C_1, ხოლო შემდეგ გამოიყენებს X-ის e_1-ს და აღადგენს M-ს:

$$C^{d_2} = C_1^{e_2 d_2} = C_1^{k \cdot \varphi(N_1)+1} = C_1 mod N_2$$

$$C_1^{e_1} = M^{d_1 e_1} = M mod N_1.$$

სხვა მომხმარებელი ვერ წაიკითხავს დაშიფრულ ტექსტს იმის გამო, რომ არ იცის d_2(e_1 იცის), ხოლო ვერ გაგზავნის X-ის სახელით ყალბ ინფორმაციას იმიდ გამო, რომ არ იცის d_1, რომელიც საჭიროა დასაშიფრად.

3. ელგამალის ალგორითმი [16].

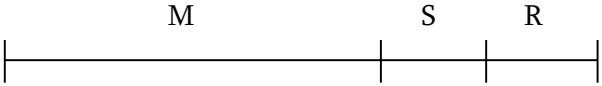
ალგორითმი გამოიყენება ღია ტექსტის ელექტრონული ხელმოწერისთვის. ტექსტი არ დაიშიფრება, მაგრამ მას დაემატება ხელმოწერა (რომელიც ორობით კოდს წარმოადგენს) და მიმღები შეძლება ტექსტის ჭეშმარიტების დადგენას.

ძირითადი ტოლობაა: $a^s = a^{xR+kM} modp$, სადაც M-ინფორმაციაა (ორობითი მიმდევრობა), p-მარტივი რიცხვი. მისი განზომლება განსაზღვრავს შემოწმების ღია კოდის განზომილებას. k-ერთჯერადი სემთხვევითი (ფსევდოშემთხვევითი) მთელი დადებითი რიცხვია, x- X-ის საიდუმლო გასაღები.

M-ტექსტის დაშიფრა არ ხდება. მას მიეწერება S და R, რომლებიც შემდეგნაირად გამოითვლება:

$R = a^k modp, S = (xR + kM) mod(p-1)$, სადაც y, p და a წინასწარაა ცნობილი (a მთელი დადებითი რიცხვია a>1): $y = a^x modp$.

გადაცემული ინფორმაცია ღებულობს შემდეგ სახეს:



M||S||R (A||B||C წარმოადგენს A,B,C სიტყვების თანმიმდევრულ შეთავსებებს- კონკატენციას.)

საზოგადოდ, მოცემული სიდიდეები მთელი დადებითი რიცხვებია, წარმოდგენილი GF(2) ველზე განსაზღვრული ვექტორთა სახით.

Y-ისთვის (და Z-ისთვისაც) ცნობილია a,y,R,M და S.

Y ხელმოწერის სისწორეს ამოწმებს ტოლობით:

$$a^s = y^R \cdot R^M modp$$

$$a^s = a^{xR+kM} = (a^x)^R \cdot (a^k)^M = y^R R^M (modp)$$

Y-მა, ერთი მხრივ, a უნდა აახარისხოს S ხარისხად a^S, მეორე მხრივ, უნდა გამოთვალოს $y^R.R^M$. თუ შესრულდა ზემოაღნიშნული პირობა, ე.ი. ხელმოწერა სწორია.

ცხრილში მოცემულია DES და RSA კრიპტოალგორითმების მახასიათებლების შედარება.

მახასიათებელი	DES	RSA
შიფრაციის სიჩქარე	მაღალი	დაბალი
გამოყენებული სიჩქარე	გადანაცვლება და ჩასმა	ხარისხში აყვანა
გასაღების სიგრძე	56 ბიტი	500 ბიტზე მეტი
კრიპტოანალიზის სირთულე (იგი განსაზღვრავს ალგორითმის მედეგობას)	გასაღების სივრცეში მთლიანი გასარჩევა	მამრავლებად დაშლა
გასაღების გენერაციის დრო	მილიწამები	წუთები
გასაღების ტიპი	სიმეტრიული	ასიმეტრიული

ამ ცხრილიდან ჩანს, რომ DES ალგორითმი ბევრად უფრო სწრაფქმედი, მოხერხებული და ადვილად რეალიზებადია, ვიდრე RSA. RSA-ს გამოყენების რთულდება ალგორითმი, მაგრამ წყდება ღია არხით ინფორმაციის გადაცემის პრობლემა.

თავი II

კოდირების ალგორითმი (შეცდომამედეგი)

სისტემური სტრუქტურა

კოდირების თეორია (შეცდომების გამსწორებელი

კოდური სისტემები), როგორის შესავალში იყო აღნიშნული, ინფორმაციის თეორიაში ფუნდამენტული შრომების [1-5] შედეგად ჩაისახა და მის ერთ-ერთ ძირითად სამეცნიერო-ტექნიკურ მიმართულებას წარმოადგენს. თეორიული თვალსაზრისით იგი ეყრდნობა მათემატიკის ისეთ დარგს, როგორიცაა ალგებრა, რიცხვთა თეორია, კომბინატორიკა, გრაფთა თეორია და კომბინატორული ალგორითმები.

ძნელია თანამედროვე დიდი ინფორმაციული კომპიუტერული სისტემების დასახელება (ინტერნეტი, ქსელური თუ სხვა ავტომატური მიზნობრივი სისტემები), რომელთა იერარქიის გარკვეული საფეხურები არ მოიცავენ კოდირების სტრუქტურებს და სისტემებს. მათდამი ინტერესი და მნიშვნელობა მხოლოდ ზოგიერთი ნაშრომისა და მონოგრაფიის დასახელებაც აშკარად ჩანს [1,3-9,19].

კოდირების ალგებრულ თეორიაში განსაკუთრებული მნიშვნელობა აქვს ვექტორულ ქვესივრცეთა ბაზისური მატრიცების სტრუქტურულ თვისებებს. მათ გამოყენებას ეფუძვნება ორიგინალური მეთოდის სინთეზი, რომელიც განხილულია ნაშრომის III თავში. მოქმედებანი GF(q) სასრულ ველზე განსაზღვრულ $V_{n,q}$ სივრცეში (კერძოდ ვექტორული და მატრიცული გარდაქმნები) წარმოადგენენ თანამედროვე ინფორმაციული სისტემების მოდელირების ერთ-ერთ მნიშვნელოვან შემადგრნრლ რგოლს. ამასთან, ნაშრომის III

თავში განხილული კრიპტოგრაფიული სისტემა ძირითადად ეფუძნება ზემოაღნიშნულ გარდაქმნებს. ამდენად, II თავში მოცემულია კოდირების ზოგიერთი ალგორითმული სტრუქტურის თვისებები.

ინფორმაციული სიტყვა

$$a = (a_1, a_2, a_3 \dots a_n) \in V_n \qquad (2.1)$$

არის n- განზომილებიანი ვექტორი; V_n სიმრავლე სალიას GF(q) ველზე განსაზღვრული n- განზომილებიანი ვექტორული სივრცეა;

$q=p^m$ -მარტივი რიცხვის ხარისხი (ემდგომში ზოგადობის დაურღვევლად შეიძლება მივიჩნიოთ, რომ q=2).

განვიხილოთ ვექტორულ სივრცეთა ზოგიერთი თვისება. ყოველი V_n ვექტორული სივრცისთვის არსებობს სივრცის ბაზისი- წრფივად დამოუკიდებელ $b^{(i)} = (b_1^{(i)}, \dots b_n^{(n)}) \in V_n$ ვექტორთა $\{b^{(1)},\dots b^{(n)}\}$ (i=1, . . . n) სიმრავლე, რომლის სტრიქონთა კომბინაციებით $a = a_1 b^{(1)} + \cdots + a_n b^{(n)} \in V_n$ (2.2)

მიიღება V_n სივრცე ; აქ $a_i (i=1, \dots n)$ GF(2) ველის ელემენტია.

ვთქვათ, V ? V_n არის V_n სივრცის k განზომილების ქვესისტემა. მაშინ არსებობს V ქვესივრცის ბაზისური მარტიცა,

$$G = \begin{bmatrix} g_{11}\, g_{12} & \cdots & g_{1n} \\ g_{21}\, g_{22} & \cdots & g_{2n} \\ \cdots \quad \cdots & & \\ g_{k1}\, g_{k2} & \cdots & g_{kn} \end{bmatrix} \qquad (2.3)$$

რომლის სტრიქონთა სივრცე (სტრიქონთა წრფივი კომპინაციები) k განზომილების ქვესივრცეა. V ქვესივრცის ნულვანი V′ სივრცე r = n − k განზომილებისაა, რომლის ბაზისი არის $G = \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1n} \\ h_{21} & h_{22} & \cdots & h_{2n} \\ \cdots & \cdots & & \\ h_{r1} & h_{r2} & \cdots & h_{rn} \end{bmatrix}$ (2.4)

ე.ი. სამართლიანია ტოლობა:

$$GH^T=0. \quad (2.5)$$

სადაც H^T არის H მატრიცის ტრანსპონირებული მატრიცა. (2.5)-გამოსახულებიდან გამომდინარეობს, რომ ნებისმიერი $a \in V$

$$aH^T=0. \quad (2.6)$$

ამასთან, ნებისმიერი b ∈ V′ ვექტორისათვის

$$bG^T=0. \quad (2.7)$$

სადაც G^T არის G მატრიცის ტრანსპონირებული მატრიცი. ე.ი. V′ ქვესივრცის ნულოვანი სივრცე არის V :

$$HG^T=0. \quad (2.8)$$

განსაზღვრება 2.1. k განზომილების V ქვესივრცეს ეწოდება წრფივი (n,k)-კოდი, თუ მისთვის G და H, შესაბამისად, მაწარმოებელი და შემომაწმებელი მატრიცებია.

V კოდისთვის H მატრიცის შემამოწმებელი ეწოდება, რადგან (2.6) გამოსახულება არის ნებისმიერი a∈V ვექტორის V ქვესიმრავლისთვის მიკუთვნების აუცილებელი და საკმარისი პირობა.

იმ შემთხვევაში, თუ:

$$aH^T \neq 0. \qquad (2.8')$$

უნდა მივიჩნიოთ, რომ $a \notin ?V$. საზოგადოდ:

$$aH^T = S. \qquad (2.9)$$

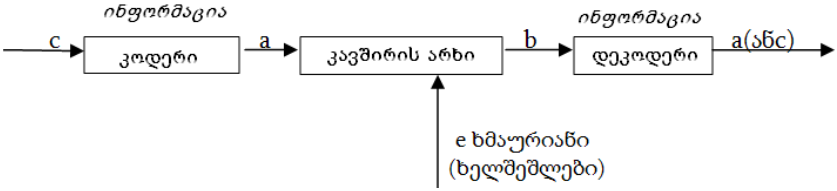
სადაც $S=(S1, \ldots S_r)$-r განზომილების ვექტორი, ანუ სინდრომია (მაშასადამე, თუ $S=0$, $a \in V$ და თუ $S \neq 0$, $a \notin ?V$).

განვიხილოთ V_n/V ფაქტორჯგუფი. (2.9) პირობა ცალსახად განსაზღვრავს {a} ფიქსირებული მოსაზღვრე კლასის ვექტორების სიმრავლეს, რადგან ყოველ მათგანს და მხოლოდ მათ V_n/V ფაქტორჯგუფებში, (2.9) პირობის თანახმად, შეესაბამება ერთი და იგივე სინდრომი, ე.ი. თუ $a' \in \{b\}$ და $b' \in \{b\}$ სხვადასხვა მოსაზღვრე კლასების ნებისმიერი ელემენდებია, მაშინ:

$$a'H^T \neq b'H^T \qquad (2.10)$$

H მატრიცის გარკვეული სტრუქტურის აგებით განისასღვრება V კოდი და განხორციელდება საჭირო V_n/V პაქტორიზაცია, ე.ი. მატრიცი შეიძლება წარმოადგენდეს ინფორმაციული სისტემების შემადგენელ ნაწილს.

განვიხილოთ კოდირების სისტემის (სიგნალების გადაცემის) მატრიცული სქემა (სურ. 2.1).



სურ. 2.1

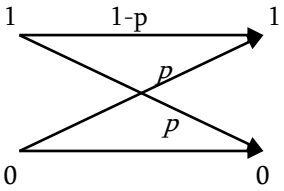
სისტემის შესვლაზე ინფორმაციის წყაროდან მიეწოდება k განზომილების ინფორმაციული ვექტორი, ანუ შეტყობინება - c-(c_1,. . . c_k). კოდი c ვექტორს გარდაქმნის $a=(a_1, \ldots a_n) \in V$ ვექტორად, რომელსაც შეუძლია გაასწოროს შეცდომები.

არხში მოქმედი ხმაურის (ხელშეშლების)ზეგავლენით მოსალოდნელია c ვექტორის სახეცვლილება (დამახინჯება), რის შედეგად არხის მეორე ბოლოზე ვღებულობთ

$$b=a+e \qquad (2.11)$$

ვექტორს, სადაც $b=(b_1, \ldots b_n)$ არის ვექტოროი $b=(a_1+e_1, \ldots a_n+e_n)$, ხოლო $e=(e_1, \ldots e_n)$ შეცდომის ვექტორიაქ. შეკრების ოპერაცია განხილულია იმ ველში, რომელზედაც განსაზღვრულია a და e ვექტორები (ორობით შემთხვევაში იგულისხმება GF(2) ველი).

განვიხილოთ ინფორმაციის გადაცემის პროცესი ხელშეშლის შემთხვევაში. სქემატურად ორობითი სიმეტრიული არხიმოცემულია სურ. 2.2-ზე.



სურ.2.2.

ორობითი სიმეტრიული არხისთვის განსაზღვრავენ სიმბოლოს (ანუ a ვექტორის $a_i \in GF(2)$ (i=1, . . . n) კომპონენტს) დამახინჯების p ალბათობას. ალბათობა იმიას რომ დეკოდერის შესავალზე მიღებული სიმბოლო თანხვდება გადაცემული სიმბოლოს, q=1-p სიდიდის ტოლია; ამრიგად, კავშირის სიმეტრიული არხი, ორობითი თანმიმდევრობით (ანუ ვექტორით) შესავალსა და გამოსავალზე, ისეთი არხია, რომლის ყოველი a_i სიმბოლო მიმღებ მხარეზე მიღება

უცვლელად (დაუმახინჯებლად) ფიქსირებული 1-p ალბათოვით და სახეს იცვლის (მახინჯდება) P ალბათობით. კავშირისა ასეთარხს უეწოდებენ არხს მეხსიარების გარეშე ან არხს დამოუკიდებელი შეცდომებით. ცხადია, (2.6) და (2.11) - დან გამომდინარეობს, რომ

$$bH^T=eH^T=s \qquad (2.11')$$

განსაზღვრება 2.2. a (2.1) ვექტორის w(a) წონა a ვექტორია არანულოვანი კომპონენტების რაოდენობის ტოლია. GF(2) ველზე განსაზღვრული ვექტორია წონა (ორობითი შემთხვევა)

$$W(a) = \sum_{i=1}^{n} a_i \qquad (2.12)$$

განსაზღვრება 2.3. ჰემინგის d(a,b) მანძილი a და b ვექტორებს შორის იმ ერთსახელა კომპონენტების (a_i,b_i) წყვილების რაოდენობაა, რომელთათვისაც $a_i \neq b_i$ (i=1, . . . n). GF(2) ველზე განსაზღვრული ვექტორისთვის (ორობითი შემთხვევა)

$$d(a,b) = \sum_{i-1}^{n}(a_i + b_i) \; . \qquad (2.13)$$

შევნიშნოთ რომ (2.13) გამოსახულების (a_i,b_i) წყვილებისთვია განსაზღვრულია შეკრებია ოპერაცია GF(2) ველში. ამიტომ აქ და შემდგომში, ადაც ეს არ იწვევს გაუგებრობას, არ ვიხმართ მოდულით 2 შეკრების აღმნიშველ "?" სიმბოლოს.

განსაზღვრება 2.4. ვთქბათ $V \subset ?V_n$ ქვესივრცის ნებისმიერი a,b $\in$ V წყვილისთვის d(a,b) ჰემინგის მანზილია, მაშინ

$$D(V)=\min d(a,b) \; ; \quad a \neq b \in V \qquad (2.14)$$

სიდიდე არის V ქვესივრცის მინიმალური მანძილი.

დავუშვათ, e (2.11) გამოსახულებაში w(e)≤t წონის ვექტორია:

$$e=b-a \qquad (2.15)$$

ე.ი. a სიტყვის არხში გადაცემის შედეგად ადგილი აქვს t - ჯერად შეცდომას.

შეცდომების გამასწორებელი კოდების თეორიის ძირითადი პრობლემა მრავალჯერადი შეცდომებისათვის შეიძლება ჩამოყალიბდეს ორი განსხვავებული სახით: 1. საჭიროა, აიგოს გარკვეული ფიქსირებული სიმძლავრის v კოდური სიმრავლე, რომლისთვისაც ყოველი მიღებული b=a+e სიტყვის შესაბამისი $a \in V$ სიტყვის აღდგენა შესალებელი იქნება e (2.15) შეცდომის შემთხვევაში t რიცხვის რაც შეიძლება მაღალი მნიშვნელობისთვის. 2. კავშირის არხში ფიქსირებული ≤t - ჯერადი e (2.15) შეცდომების არსებობის პირობებში საჭიროა, აიგოს რაც შესაძლებელი მაღალი სიმძლავრის კიდური სიმრავლე V, რომლისთვისაც შესაძლებელი იქნება ყოველი მიღებული b=a+e სიტყვის აღდგენა. თუ $V \subset ?V_n$ კოდური სიმრავლე დასმულ ამოცანას წყვეტს, მაშინ V ქვესივრცის t-ჯერადი შეცდონის გამასწორებელი წრფივი კოდი ეწოდება. V კოდურ სიმრავლეს ახასიათებს: n=k+r კოდური ვექტორებისსიგრძე, ანუ კოდური (2.1) სიტყის სიმბოლოების რაოდენობა, k - ინფორმაციულ სიმბოლოთა რაოდენობა, ანუ კოდის მაწარმოებელი (2.3) მატრიცის რანგის მნიშვნელობა, r - შემოწმებულ სიმბოლოთა რაოდენობა, ანუ კოდის შემამოწმებელი (2.4) მატრიცის რანგის მნიშვნელობა, (ასეთ

კოდს t -ჯერადი შეცდომების გამასწორებელ (მაკორექტირებელ) წრფივ (n,k) - კოდს უწოდებენ).

თუ K (2.1) სიტყვათა რაიმე სიმრავლეა, მაშინ t-ჯერადი შეცდომების გასასწორებლად საჭიროა, შესრულდეს შემდეგი პირობა: $d(K)\geq 2t+1$ (2.16)

მართლაც, (2.16) გამოსახულება ნიშნავს იმას ,რომ $a+b\in K$ ნებისმიერ წყვილს შორის მანძილი არ არის 2t+1 სიდიდეზე ნაკლები და მაშასადამე, t - ჯერადი შეცდომა შეიძლება გასწორებულ იქნას.

შეცდოების კორექტირების საკითხი წრივ V კოდში შეიძლება ასე გადაწყდეს:

იმისთვია რომ $V \subset ? V_n$ კოდი ასწორებდეს t- ჯერად შეცდომებს, საჭიროა რომ

$$W(a)\geq 2t+1 \qquad (2.17)$$

ნებისმიერი $a\in V$ არანულოვანი ვექტორისთვის.

მართლაც თუ

$$w_{min}(V) = \min_{a\neq 0\in V} w(a) = 2t+1\,, \qquad (2.18)$$

მაშინ ნებისმიერი $a\neq b\in V$ წყვილისთვის

$$d(V)=d(a,b)=w(a+b)\geq 2t+1 \qquad (2.19)$$

რადგან, თუ დავიშვებთ, რომ

$$w(a+b)<2t+1, \qquad (2.20)$$

მაშინ, v ქვესივრცის შეკრების ოპერაციის მიმართ ჩაკეტილობის გამო, მივიღეთ, რომ

$$w(c)<2t+1 \qquad (2.21)$$

სადაც $a+b=c \in V$. (2.21) გამოსახულება ეწინაარმდეგება (2.17) დაშვებას. მაშასადამე, V კოდის მინიმალური მანძილი არ არის 2t+1 სოდოდეზე ნაკლები (როდესაც სრულდება (2.17) პირობა), ე.ი.

$$d(V)\geq 2t+1 , \qquad (2.22)$$

და კოდი ასწორებს t-ჯერად შეცდომებს.

ამრიგად, ვექტორთა მინიმალური წონისა და მინიმალური მანძილის გამოყენებით შეიძლება აიგოს დამოუკიდებელი t- ჯერადობის შეცდომების გამასწორებელი კოდი, მაგრამ (2.16) და (2.17) პირობების შუალო შესწავლა დაკავშირებულია ვექტორების გადასინჯვასთან 2^k (საზოგადოდ q^k) სიმძლავრის V კოდურ სიმრავლეშებში. ეს კი კომპიუტერული რეალიზაცვიის დროს n- ისა და k-ს დიდი მნიშვნელობისთვის იწვევს პრაქტიკულად გადაულახავ სიქნელეებს. მდგომარეობიდან გამოსვლას წარმოადგენს საჭირო სტრუქტურის G (2.3) და H (2.4) მატრიცების აგება.

ვიდრე კოდების მატრიცულ აღწერას განვიხილავთ, ვისარგებლოთ შემდეგი ცნობილი განსაზღვტებით [20]:

განსაზღვრება 2.5. GF(q) ველზე განსაზღვრული V_n ვექტორული სივრცის $a^{(1)} \ldots a^{(w)}$ ვექტორების ერთობლიობას ეწოდება წრფივად დამოკიდებული, თუ

$$a^{(1)} a^{(1)} + \ldots + a^{(w)} a^{(w)} = 0 \qquad (2.23)$$

$a^{(1)} \in GF(q)(i=1, \ldots w)$ სკალარების გარკვეული მნისვნელობისთვის, როდესაც $a^{(1)}, \ldots a^{(w)}$ ელემენტები ერთდროულად არ უდრის ნულს. $a^{(1)}, \ldots a^{(w)}$ ვექტორების ერთობლიობა წრფივად დამოკიდებულია, თუ ის წრფივად დამოკიდებული არ არის.

არსებობს კავშირი (2.23) წრფივად დამოკიდებულებისა და (2.2) წრფივ კომბინაციებს შორის. მართლაც, თუ $a^{(1)}, \ldots a^{(w)}$ ვექტორთა რაიმე ერთობლიობის რომელიმე ვექროტი არის დანარჩენი ვექტორებივ წრფივი კომბინაცია, მაშინ ეს ერთობლიობა წრფივად დამოკიდებულია. (2.6) პირობიდან გამომდინარე, H მატრიცის იმ ვექტორ-სვეტის ჯამი, რომლებიც შეესაბამებიან a ვექტორის არანულოვან კომპონენტებს (2.6′) გამოსახულებაში, სწრაფად დამოკიდებულ ერთობლიობას შეადგენენ.

კოდების მატრიცული აღწერა გაცილებით უფრო კომპაქტურია V სიმრავლესთან შედარებით. კოდის საჭირო სტრუქტურის აგების თვალსაზრისით მნიშვნელოვანია შემდეგი ცნობილი თეორემა [80], რომელიც, როგორც შემდეგ პარაგრაფში არის ნაჩვენები, ახალ მნიშვნელობას იძენს m- პათეტიკური შეცდომების განხილვასთან დაკავშირებით.

<u>თეორემა 2.1.</u> V წრფივი კოდის w წონის წოველ კიდურ a ვექტორს V ქვესივრცის ნულოვანი სივრცის H მატრიცაში შეესაბამება w რაოდემობის გარკვეული სვეტების წრფივი

დამოკიდებულება და, პირიქით, ყოველ წრფივ დამოკიდებულებას, შედგენილს w სვეტისგან, V სიმრავლეში შეესაბამება w წონის ერთი გარკვეული კოდური ვექტორი.

თეორემის მტკიცება ეკრდნობა იმ ფაქტს, რომ $a=(a_1, \ldots a_n)$ [2.1] ვექტორი არის კოდური მაშინ და მხოლოდ მაშინ, როდესაც ის აკმაყოფილებს [2.6] დამოკიდებულებას. ეს უკანასკნელი შეიძლება ასე გადავწეროთ:

$$\sum_{j=1}^{n} a_j h^{(j)} = 0 \qquad (2.24)$$

სადაც $h^{(j)}=(h_1{}^{(j)}, \ldots h_n{}^{(j)})$ არის H მატრიცის j- ური ვექტორ-სვეტი. [2.24] წრფივ დამოკიდებულებაში მონაწილეობს ის ვექტორ-სვეტები, რომლებიც მასში შედის არანლოვანი a_j კოეფიციენტებით. ამრიგად, განსაზღვრება 2.2.-ის თანახმად, წეფივ დამოკიდებულებაში მონაწილეობს იმდენი ვექტორ-სვეტი, რა წონაცაა კოდური ვექტორი. მეორე მხრივ, w რაოდენობის $h^{(j)}$ ($j \in \{1,. \ldots n\}$) ვექტორ- სვეტებისგან შედგენილ წრფივ დამოკიდებულებაში a ვექტორი შედის w რაოდენობის a_j არანულოვანი კოეფიციენტით. ამიტომ w რაოდენობის ვექტორ-სვეტების ყოველ წრფივ დამოკიდებულებას შეესაბამება w წონის a კოდური ვექტორი.

როგორც თეორემა 2.1. - დან ჩანს, მნშვნელოვანია H მატრიცის სტრუქტურებისკვლევა V სივრცის საჭირო თვისებების მისაღებად. 2.1. თეორიიდან გამომდინარეობს შემდეგი შედეგი:

შედეგი 2.1 V კოდის მინიმალური წონა არის w მაშინ და მხოლოდ მაშინ, როდესაც მიდი ნულოვანი სივრცის ბაზისური H მატრიცის ნებისმიერი w-1 და ნაკლები რაოდენობის სვეტებისგან შემდგარი ერთობლიობა სრფივად დამოკიდებულია.

ამრიგად, T-ჯერადი შეცდომების გამასწორებელი V კოდისთვის სამართლიანია შემდეგი გამოსახულება:

$$e_{2t}H^T \neq 0, \qquad (2.25)$$

სადაც e_{2t} 2t-ჯერადი შეცდომის ვექტორია.

2.1. შედეგი საშუალებას იძლევა ავაგოთ კოდები, რომლებიც ორობით სიმეტრიულ არხში ასწორებენ t-ჯერად დამოუკიდებელ შეცდომებს.

თავი III

ალგებრულ (მატრიცულ) სტრუქტურებზე დაფუძნებული კრიპტოგრაფიული მეთოდის სინთეზი

§ 3.1 ზოგადი მიდგომა მატრიცული გასაღების მისაღებად

განვიხილოთმატრიცული გასაღების მეთოდი. ეს მეთოდი მდგომარეობს შემდეგში: ტექსტი, ანუ a (2.1) ორობითი სიტყვა იყოფა გარკვეული სიგრძის ბლოკებად. შიფრაცია ხდება შემდეგნაირად: ბლოკი, როგორც x ვექტორი, გამრავლდება შესაბამისი განზომილების მატრიცაზე. დეშიფრაციის დროს კი

მიღებული ვექტორი უნდა გავამრავლოთ შებრუნებულ მატრიცაზე, რაც აღადგენს საწყის ვექტორს:

$$x \cdot A = x' \;; \quad x' \cdot A^{-1} = x \qquad (3.1)$$

A მატრიცას მოეთხოვება, რომ იყოს გადაუგვარებელი (ე.ი. დეტერმინანტი 0-ის ტოლი არ უნდა იყოს), რათა გააჩნდეს შებრუნებული, ე.ი. დგას პრობლემა გადაუგვარებელი მატრიცების სინთეზისა და მათი შებრუნებულის პოვნისა. როდესაც საქმე გვაქვს დიდი განზომილების მქონე მატრიცასთან, მათი შებრუნებულის გამოთვლა ცნობილი მეთოდით გარკვეულ დროს მოითხოვს. ამიტომ საჭიროა, შეიქმნას რეგულარული მეთოდები, რომლებიც მარტივი ალგორითმებით მოგვცემენ მატრიცის შებრუნებულს. ე.ი. გამოიყოს გარკვეული კლასი მატრიცებისა, რომელთა შებრუნებულის პოვნა რეალიზდება (გარკვეული წესით) რთული ალგორითმული გამოთვლის გარეშე.

განვიხილოთ ცნობილი მეთოდები მატრიცის შებრუნებულის პოვნისას. ცნობილია არაერთი მეთოდი. შესაძლებელია $A=(a_{ij})^n$ მატრიცის შებრუნებულის პოვნა (თუ ის არასინგულარულია) შემდეგი სახით [11]:

$$A^{-1} = \begin{bmatrix} \frac{A_{11}}{|A|} & \frac{A_{21}}{|A|} \dots & \frac{A_{n1}}{|A|} \\ \frac{A_{12}}{|A|} & \frac{A_{22}}{|A|} \dots & \frac{A_{n2}}{|A|} \\ \dots & \dots & \dots \\ \frac{A_{1n}}{|A|} & \frac{A_{2n}}{|A|} \dots & \frac{A_{nm}}{|A|} \end{bmatrix} \qquad (3.1)$$

სადაც A_{ij} არის A მატრიცის a_{ij} ელემენტის ალგებრული დამატება.

როგორც ვხედავთ, მიუხედავად იმისა, რომ ორობითი GF(2) ველზე (3.1) ოპერაციების ჩატარება შედარებით ადვილია, საჭირო გამოთვლები, ცხადია, დროს მოითხოვს და, რაც მთავარია, მისი მეშვეობით ძნელია გარკვეული კლასის ფორმირება, რომლისთვისაც (3.1) მატრიცები მიიღება გრივიალური გზით.

დასახული მიზნის მიღწევა, ჩვენი აზრით, არც შედეგი მატრიცული ნამრავლის მეშვეობითაა შესაძლებელი:

$$E_k E_{k-1} \dots E_1 A=1, \qquad (3.2)$$

და

$$E_k E_{k-1} \dots E_1 =A^{-1}, \qquad (3.3)$$

სადაც A^{-1} A, მატრიცის მარცხენა შებრუნებული მატრიცაა; $E_1 \dots E_k$ წარმოადგენენ ელემენტარულ მატრიცებს, რომელთა მეშვეობით A მატრიცი შესაძლებელია დავიყვანოთ კანონიკურ და, მაშასადამე, ერთეულოვან სახამდე.

განსხვავებულ მეთოდს წარმოადგენს A^{-1} მატრიცის i-ური სვეტის $x_1, x_2, \dots x_n$ ელემენტების მისაღებად (გამომდინარე $AA^{-1}=I$ ტოლობიდან) განტოლებათა შეგეგი სისტემის ამოხსნა:

$$a_{i1}x_1 + a_{k2}x_2 + \cdots + a_{kn}x_n = \begin{cases} 0, თუ\ k \neq i \\ 0, თუ\ k = i \end{cases} \qquad (3.4)$$

სადაც k=1,2,…n. რადგან $|A|\neq 0$, ამიტომ (3.4) სისტემას აქვს ერთადერთი ამონახსნი, რის შედეგადაც, საზოგადოდ, A^{-1} შებრუნებული მატრიცი მიიღება.

კოდირეის ალგებრული თეორიიდან ცნობილია, რომ

მრავალწევრთა ალგებრაში GF(q) ველზე მოდულით f(x) მიიღება კლასები მატრიცებსა, რომლებიც წარმოქმნიან (2.3) და (2.4) სახის მაწარმოებელ და შემამოწმებელ ბაზისურ მატრიცებს, რომლებიც აკმაყოფილებენ (2.5) პირობას. შესაბამისი მატრიცების სტრიქონთა სივრცე წარმოადგენს მრავალწევრთა იდეალებს. ასეთი მატრიცებისათვის დამახასიათებელია g(x) და h(x) (g(x)·h(x)=f(x))მწარმოებელი მრავალწევრები, რომლებიც G (2.3) და (2.4) ბაზისური მატრიცების სტრიქონებს აფორმირებენ [8] (რაც განხილულია 3.2).

აღნიშნულის ანალოგიურად n რიგის კვადრატული მატრიცები და მათი შებრუნებულები შესაძლოა შავწეროთ შემდეგი სახით:

$$A = \begin{bmatrix} a_1 & a_1 & a_3 & \cdots & a_{n-1} & a_n \\ 0 & a_1 & a_2 & \cdots & a_{n-2} & a_{n-1} \\ 0 & 0 & a_1 & \cdots & a_{n-3} & a_{n-2} \\ & & \cdots\cdots\cdots & & & \\ 0 & 0 & 0 & & a_1 & a_2 \\ 0 & 0 & 0 & \cdots & 0 & a_1 \end{bmatrix}, \qquad (3.5)$$

სადაც (3.5) მატრიცის სტქირონებს მრავალწევრთა იდეალების ბაზისური მატრიცების მსგავსად შეადგენენ GF(2) ველზე განსაზღვრული a∈ V_n ვექტორის კომპონენტები (ანუ A მატრიცს აწარმოებს g=(a_1,… a_n) ვექტორის კომპონენტებით).

ფიქსირებული ვექტორისთვის შესაძლებელია შებრუნებულის პოვნის ერთ-ერთი მეთოდით ((3.4) სისტემის ეშვეობით) განსაზღვროს h ვექტორის სახე ნებისმიერი მთელი

დადებითი n-ისათვის. მაგალითად, (3.4)-ისა და მათემატიკური ინდუქციის გამოყენებით შეიძლება ვაჩვენოთ, რომ n რიგის

$$A = \begin{bmatrix} 1 & 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & 1 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ . & . & . & . & . & . \\ 0 & 0 & 0 & \dots & 1 & 1 \\ 0 & 0 & 0 & \dots & 0 & 1 \end{bmatrix} \qquad (3.6)\leq$$

მატრიცისთვის, რომლის მაწარმოებელი ვექტორი არის

$$g = (a_1, a_2, \dots a_n)\big(a_i = 1 \text{ თუ } i \leq 2 \text{ და } a_i = 0, \text{ თუ } i > 2\big),$$

შებრუნებულ მატრიცს აქვს სახე:

$$A^{-1} = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 & 1 \\ 0 & 1 & 1 & \dots & 1 & 1 \\ 0 & 0 & 1 & \dots & 1 & 1 \\ . & . & . & . & . & . \\ 0 & 0 & 0 & \dots & 1 & 1 \\ 0 & 0 & 0 & \dots & 0 & 1 \end{bmatrix}, \qquad (3.7)$$

სადაც $h=a_1, \dots a_n)$, $a_i=1 (1\leq i\leq n)$.

ანალოგიურად, $g=(a_1,a_2, \dots a_n)$ $(a_1=1, i\leq 3; a_i=0, i>3)$ და $h=(a_1,a_2, \dots a_n)$ $(a_i=1 , i=3k+1, i=3k+2; a_i=0, i=3k)$ მაწარმოებელი ვექტორისთვის შესაბამისად მიიღება:

$$A = \begin{bmatrix} 1 & 1 & 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & 1 & 1 & \dots & 0 & 0 \\ 0 & 0 & 1 & 1 & \dots & 0 & 0 \\ 0 & 0 & 0 & 1 & \dots & 0 & 0 \\ . & . & . & . & . & . & . \\ 0 & 0 & 0 & 0 & \dots & 1 & 1 \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 \end{bmatrix}, \quad A^{-1} = \begin{bmatrix} 1 & 1 & 0 & \dots & 1 & 1 & 0 \\ 0 & 1 & 1 & \dots & 0 & 1 & 1 \\ 0 & 0 & 1 & \dots & 1 & 0 & 1 \\ 0 & 0 & 0 & \dots & 1 & 1 & 0 \\ . & . & . & . & . & . & . \\ 0 & 0 & 0 & \dots & 0 & 1 & 1 \\ 0 & 0 & 0 & \dots & 0 & 0 & 1 \end{bmatrix} \qquad (3.8)$$

ე.ი (3.8) მატრიცებს აწარმოებს შესაბამისად g და h ვექტორები იმ პირობით, რომ მატრიცებში ყოველი i-ური სტრიქონი წარმოადგენს (i-1)-ე სტრიქონის კომპონენტების გადანაცვლებას ერთი პოზიციით და რომ (3.8) მატრიცის ყოველი (i+1)-ე სტრიქონის პირველი i კომპონენტი ნულის ტოლია.

მოცემული (3.6) – (3.8) სახის მატრიცების სინთეზის მიზანი მდგომარეობს იმაში, რომ არ განვახორციელოთ მატრიცის შებრუნებულის გამოთვლა, არამედ გარკვეული მარტივი შესაბამისობით შეგვეძლოს შებრუნებულის პოვნა.

იმისთვის, რომ გასაღებს გააჩნდეს მაღალი მედეგობა, საჭიროა, სიმრავლეში იყოს საკმაოდ დიდი რაოდენობის გასაღები. ავიღოთ n-განზომილებიანი ერთ-ერთი განხილული მატრიცა და მისი შებრუნებული. თუ საწყის მატრიცაში გადავაადგილებთ სტრიქონებს, მიღებული მატრიცის შებრუნებულის საპოვნელად საჭიროა, საწყისი მატრიცის შებრუნებულში გადავაადგილოთ შესაბამისი სიტყვები, ე.ი. ერთი მატრიციდან შეგვიძლია მივიღოთ n! რაოდენობის მატრიცა. ასევე შესაბამისი სვეტებისა და სტრიქონების

გადანაცვლებით მიიღება ისეთივე რაოდენობის მატრიცა და მთლიანობაში $(n!)^2$ სიმრავლე ფიქსირებული g და h ვექტორებისათვის.

მატრიცული გასაღების მეთოდით საჭირო გასაღების სინთეზი შესრულდება შემდეგი თანმიმდევრობით: შეირჩევა გარკვეული g და h ვექტორები, მოხდება A da A^{-1} მატრიცების გენერაცია დაშემდეგ შემთხვევითი (ფსევდოშემთხვევითი) რიცხვის შესაბამისად მატრიცებში განხორციელდება სათანადო გადანაცვლებები.

რა უპირატესობა გააჩნია მატრიცულ მეთოდს ვიჟინერის მეთოდთან შედარებით, სადაც ტექსტი ასევე ბლოკებად იყოფა? ვიჟინერის მეთოდში რომ „გატყდეს“ ერთ-ერთი ბლოკის ნაწილი, ცნობილი გახდება თვითონ გასაღების ნაწილი და, შესაბამისად, ეს ნაწილი „გატყდება“ ყოველ ინფორმაციულ ბლოკში; ხოლო მატრიცული გასაღების დროს ერთ ბლოკში ინფორმაციის გახსნით შეუძლებელია მატრიცის პარამეტრების მიღება და მთლიანად გასაღების „გატეხვა“.

ნაშრომში მატრიცული გასაღებების მეთოდი გამიზნულია კომბინირებული კრიპტოსისტემის შესაქმნელად, რომელშიც ერთად იქნება გამოყენებული ღია არხით სარგებლობის ცნობილი მეთოდები და მატრიცული მეთოდი, თუმცა გარკვეული მომხმარებლისათვის მისი არც ცალკე (სხვა მეთოდებისგან დამოუკიდებლად) გამოყენებაა მიღებული.

§3.2.კრიპტოგრაფიული გასაღებების სინთეზი მრავალწევრთა ალგებრაში

განხილული ამოცანის გადაწყვეტა უფრო მიზანდასახული იქნება, თუ გამოვიყენებთ კოდირების ალგებრულ სტრუქტურებს, კერძოდ, GF(q) ველზე (სიმარტივისთვის განვიხილავთ GF(2)ველს) მოდულით f(x) მრავალწევრთა ალგებრაში იდეალების თვისებებს.

ცნობილია, რომ n-განზომილებიან მრავალწევრთა ნაშთთა კლასები მოდულით f(x) GF(2) ველზე წარმოქმნიან მრავალწევრთა A_n ალგებრას და, მაშასადამე, ვექტორულ V_n სივრცეს (ბგულისხმობთ, რომ

$a=(a1,\dots an) \in Vn$ და $a(x) = \sum_{i=0}^{n} a_i x^i \in A_n$

წარმოადგენენ ექვივალენტურ ელემენტებს).

ცნობილია, ასევე, რომ A_n ალგებრაში ნებისმიერი I იდეალისთვის არსებობს ერთადერთი ნორმალური g(x) მრავალწევრი მინიმალური ხარისხისა ისეთი, რომ {g(x)} ნაშთთა კლასი ეკუთვნის I იდეალს და პირიქით, თითოეული ნორმირებული g(x) მრავალწევრი, გამყოფი g(x)-ისა, აწარმოებს გარკვეულ I იდეალს, რომელშიც g(x) არის მინიმალური ხარისხის მრავალწევრი ისეთი, რომ g(x) ნაშთთა კლასი ეკუთვლის I იდეალს.

სამართლიანია შემდეგი

<u>თეორემა 3.1.</u> ვთქვაქთ,f(x)=g(x)h(x), სადაც f(x) არის n

ხარისხის მრავალწევრი, ხოლო h(x)-k ხარისხისა. მაშინ {g(x)} ნაშთთ კლასით ნაწარმოები იდეალი მოდულით f(x) მრავალწევრთა ალგებრაში არის კი განზომილებისა.

ეს ნიშნავს, რომ g(x) მრაცალწევრის ხარისხი არის

$$n-k=r \tag{3.9}$$

სამართლიანია აგრეთვე

თეორია 3.2. ვთქვათ, f(x)=g(x) და h(x) ნორმირებული მრავალწევრებია და f(x)=g(x)h(x), მაშინ {a(x)} ნაშთთა კლასი ეკუთვმის h(x)-ით ნაწარმოები იდეალის ნულოვან სივრცეს მაშინ და მხოლოდ მაშინ, როდესაც ის ეკუთვნის g(x) -ით ნაწარმოებ იდეალს.

ზემოთქმულიდან გამომდინარეობს შემდეგი

შედეგი 3.1. ვთქვათ, f(x)= g(x)h(x), სადაც f(x)-n ხარისხის და g(x)-r ხარისხის მრავალწევრებია. მაშინ

$$GH^T=0,$$

სადაც G და H მატრიცებს შესაბამისად g(x) და h(x) მრავალწევრები აწარმოებენ.

$g=(g_0, \dots g_{n-1})$ ვექტორის კომპონენტების ციკლური გადანაცვლება i პოზიციით წარმოადგენს $g(i)=(g_i \dots g_{n-1})$ ვექტორს; ანუ $g(x)=1+x+ \dots + x^r$ მრავალწევრის i-ურო გადანაცვლება გვაძლებს $g(x^{(i)})=x^i g(x) \bmod (x^n-1)$ მრავალწევრს.

ვთქვაქთ $g(x)h(x)=x^n-$, 1 g(x) და h(x) აწარმოებენ შესაბამისად I და I' იდეალებს. მაშინ

$$G = \begin{bmatrix} g_0\ g_1 \dots & g_r & 0 \dots & 0 \dots 0 \\ 0 & g_0 \dots & g_{r-1}\ g_r \dots & 0 \dots 0 \\ \dots \dots \dots & \dots \dots \dots & & \dots \dots \dots \\ 0 & 0 \dots & 0 & 0 \dots & g_0 \dots g_r \end{bmatrix}, \qquad (3.10)$$

$$H = \begin{bmatrix} h_0^* & h_1^* \dots & h_k^* & 0 \dots & 0 \dots 0 \\ 0 & h_o^* \dots & h_{k-1}^* & h_k^* \dots & 0 \dots 0 \\ \dots \dots \dots & \dots \dots \dots & & \dots \dots \dots \\ 0 & 0 \dots & 0 & 0 \dots & h_0^* \dots h_k^* \end{bmatrix} \quad (3.11)$$

რაც ნიშნავს, რომ ნებისმიერი $g(x^{(i)})$ და $h(x^{(j)})$ მრავალწევრებისთვის სამართლიანია ტოლობა:

$$g(x^{(i)})\ h(x^{(j)}) \equiv 0 \bmod (x^n - 1) \qquad (3.12)$$

სადაც I,jЄ {1,…n}. თუ გავითვალისწინებთ, რომ GF(2) ველზე მრავალწევრთა და ვექტორთა ნამრავლი არ ემთხვევა ერთმანეთს, ნაშინ ნებისმიერი gϵI ვექტორისთვის

$$gH^{*T} = 0, \qquad (3.13)$$

სადაც H^* მატრიცი ნაწარმოებია h^* ვექტორით, რომელიც შეიცავს h ვექტორის კომპონენტებს, ჩაწერილს საწინააღმდეგო თანმიმდევრობით (ე.ი.h^* წარმოადგენს h ვექტორის სარკისებულ შებრუნებულს).

მაშასადამე, (3.12) და (3.13) ექვივალენტური ტოლობები (რაც ჩვენთვის მნიშვნელოვანია) სამართლიანია, რადგან I და I^* იდეალები წარმოადგენენ ჩაკეტილ სიმრავლეებს ვექტორთა ნებისმიერი ციკლური წანაცვლების მიმართ.

განვიხილოთ (3.5) მატრიცის შესაბამისი n რიგის

კვადრატული მატრიცები, ნაწარმოები g(x) და h(x) მრავალწევრებით (რომელთა კოეფიციენტების მეშვეობით მიღებულია (3.10) და (3.11) მატრიცების სტრიქონები):

$$A_1 = \begin{bmatrix} g_0 & g_1 & \dots & g_r & 0 & \dots & 0 \\ 0 & g_0 & \dots & g_{r-1} & g_r & \dots & 0 \\ . & . & . & . & . & . & . \\ 0 & 0 & \dots & 0 & 0 & \dots & g_0 \end{bmatrix}, \quad (3.14)$$

$$A_2 = \begin{bmatrix} h_0 & h_1 & \dots & h_k & 0 & \dots & 0 \\ 0 & h_0 & \dots & h_{k-1} & h_k & \dots & 0 \\ . & . & . & . & . & . & . \\ 0 & 0 & \dots & 0 & 0 & \dots & h_0 \end{bmatrix}, \quad (3.14')$$

სადაც A_2 მატრიცის ყოველი j-ური სვეტი წარმოადგენს h'(j) ვექტორს მოდულით x^n-1 ალგებრაში, რომლის i-ური კონპონენტები იგივეა, რაც $h^*(x)x^{r+j-1}$ ვექტორის კონპონენტები, თუ $i \le j$ და $h'_i=0$, თუ $i>j$. ყოველივე ზემოთქმულიდან ((3.9) პირობის გათვალისწინებით) გამომდინარეობს რომ

$$g(i)h'(j)^T=\begin{cases} 0, & i \neq j \\ 1, & i \neq j \end{cases}. \quad (3.15)$$

სადაც h'^T ვექტორი (3.15) ტოლობაში წარმოადგენს ვექტორ-სვეტს (ანუ h' ვექტორის ტრანსპონირებულ ვექტორს)

მაშასადამე, სამართლიანია

<u>თეორემა 3.1.</u> ვთქვათ, g(x) და h(x), შესაბამისად, r და k ხარისხის მრავალწევრებია GF(2) ველზე მოდლით (x^n-1) ალგებრაში ისეთი, რომ $g(x)h(x)=x^n-1$, ხოლო A_1 და A_2 n რიგის g(x) და h(x) მრავალწევრებით ნაწარმოები მატრიცაა (3.14) . მაშინ A_1 და A_2 ურთიერთშებრუნებულია, ე.ი.

$$A_1 A_2 = I, \quad A_2 A_1 = I,$$

სადაც I ერთეულოვანი მატრიცაა.

არსებობს კონსტრუქციული მეთოდი x^n-1 მოდულით ალგებრაში $g(x)$ და $h(x)$ მრავალწევრების მიღებისა, რომელთათვისაც $g(x)h(x)=x^n-1$. რაც **3.1.** თეორემით მიღებული მეთოდის კონსტინუქციული განხორციელებისთვის საჭირო წინაპირობებს უზრუნველყოფს.

მაგალითი. მრავალწევრთა A_7 ალგებრაში მოდულით x^7-1 $GF(2)$ ველზე $g(x)=1+x+x^3$ და $h(x)=1+x+x^2+x^4$ მრავალწევრებისათვის $g(x)h(x)=x^7-1$. მიიღება ურთიერთშებრუნებული მატრიცები.

$$A_1=\begin{bmatrix}1&1&0&1&0&0&0\\0&1&1&0&1&0&0\\0&0&1&1&0&1&0\\0&0&0&1&1&0&1\\0&0&0&0&1&1&0\\0&0&0&0&0&1&1\\0&0&0&0&0&0&1\end{bmatrix},\ A_2=\begin{bmatrix}1&1&1&0&1&0&0\\0&1&1&1&0&1&0\\0&0&1&1&1&0&1\\0&0&0&1&1&1&0\\0&0&0&0&1&1&1\\0&0&0&0&0&1&1\\0&0&0&0&0&0&1\end{bmatrix} \qquad (3.16)$$

მე-4 თავში განხილულია 3.1 თეორემით მიღებილი მატრიცების ფორმირებისთვის ალგორითმები და კომბინირებული კრიპტოგრაფიული სისტემის მოდელირების საკითხები.

თავი IV

კრიპტოგრაფიული მეთოდის სინთეზი. მეთოდის ალგორითმული და პროგრამული განხორციელება

§1. კრიპტოგრაფიული სისტემის აღწერა

გადამცემი დამიმღები ახდენენ გასაღების სინთეზს, რომელიც იმუშავებს გარკვეული დროის მანძილზე (რამდენიმე დღიდან რამდენიმე თვემდე - მედეგობის მიხედვით). ამ ხნის განმავლოვაში ყველა ინფორმაცის იშიფრება ამ გასაღებით. გასაღები წარმოადგენს n-განზომილებიან მატრიცას და მის შებრუნებულს. მისი მედეგობა დამოკიდებულია n რიცხვის სიდიდეზე.

გასაღების სინთეზი ხდება შემდეგნაირად: შექმნილია ბაზა, რომელშიც ჩაწერილია სხვადასხვა n-თვის $g(x)$ და $h(x)$ მრავალწევრები GF(2) ველიდან ისეთი, რომ $g(x)h(x)=0 \bmod f(x)$, სადაც $f(x)=x^n-1$. $g(x)$ და $h(x)$ მრავალწევრების კოეფიციენტები წარმოადგენენ შესაბამისად g და h ვექტორების კომპონენტებს. ამ ვექტორებიდან მიიღება მატრიცები, რომელთა პირველი სტრიქონები წარმოადგენენ მოცემულ ვექტორებს, ხოლო

ყოველი შედაგი სტრიქონი მიღებულია წინა სტრიქონის ერთი ნაბიჯით მარჯვნივ წანაცვლებით (მარცხნიდან ემატება 0).

გადამცემი დამიმღები ერთობლივად - გასაღებების ღია არხით გაცვლის რომელიმე მეთოდით - ახდენენ 3 რიცხვის სინთეზს: ერთი მათგანი მოახდენს ბაზიდან g(x) და h(x) მრავალწევრების შერჩებას, დანარჩენი ორის მიხედვით კი შეირჩება 1,2, ... n რიცხვების გარკვეული გადანახვლება: შემდეგ ერთი გადანაცვლების შესაბამისად დამშიფრავ მატრიცაში მოდხა სტრიქონების გადაადგილება, ხოლო გამშიფრავ მატრიცაში - სვეტების, მეორე გადანაცვლების შესაბამისად კი მიღებულ დამშიფრავ მატრიცაში მოხდება სვეტების გადანაცვლება, ხოლო გამშიფრავ მატრიცაში - სტრიქონების. მიიღება G დამშიფრავი და H გამშიფრავი მატრიცები.

გადამცემს გასაგზავნი ინფორმაცია გადაყავს ორობით მიმდებრობაში, რომელსაც ყოფს გასაღების (მატრიცის) განზომილების ტოლ ბლოკებად. შემდეგ თტოეულ ბლოკს მატრიცულად ამრავლებს G მატრიცაზე და მიღებულ ორობით მიმდებრობას უგზავნის მიიმღებს. მიმღები ამ მიმდევრობას ყოფს მატრიცის განზომილების ტოლ ბლოკებად და თითოეულ ბლოკს მატრიცულად ამრავლებს H მატრიცაზე. მიიღება საწყისი მიმდევრობა, რომლის მიხედვითაც იგი აღადგენს საწყის ტექსტს.

§2. პროგრამული რეალიზაცია ეგმ-ზე და მაგალითი ექსპარიმენტისთვის

მოდელი შედგება სამი ნაწილისაგან; 1. გასაღების სინთეზი; 2. შეტანილი ინფორმაციის დაშიფრა; 3. დაშიფრული ინფორმაციიდან საწყისი ინფორმაციის აღდგენა. მოდელისთვის (როგორც კერძო მაგალითი) აღებულია 8- განზომილებიანი მატრიცა, ხოლო ტექსტი შეიცავს 5 სიმბოლოს. (ასო-ნიშანს).

1. გასაღების სინთეზი.

თავიდან მოცემული გვაქვს ორი 8-განზომილებიანი ვექტორი, რომლებიც წარმოადგენენ $g(x)$ და $h(x)$ მრავალწევრების კოეფიციენტებს(ან ჩვენ შეგვიძლია ეს კოეფიციენტები თვითონ შევიტყოთ). ამ ვექტორების კომპონენტების მიხედვით იქმნება მატრიცები. ჩვენ შეგვყავს რიცხვი 1-დან 8!=40320 ფარგლებში. ამ რიცხვის მეშვეობით ხდება 1,2, ... n რიცხვების გარკვეული გადანაცვლების სინთეზი. ამ გადანაცვლების შესაბამისად პირველ მატრიცაში ხდება

სტრიქონების გადანაცვლება, ხოლო მეორეში-სვეტების. ასე მიიღება G დამშიფრავი და H გამშიფრავი მატრიცები.

2. ინფორმაციის (ტექსტის) დასიფვრა.

მეორე ნაწილში ჩვენ შეგვყავს ტექსტი, რომელიც უნდა დაიშიფროს (5 სიმბოლო). ჯერ ხდება ამ სიმბოლოების ASCII-კოდებში გადაყვანა, ხოლო შემდეგ- კოდების 8-თანრიგიან ორობით რიცხვებად გადაქცევა. თითოეული რიცხვი, როგორც ორობითი ვექტორი, მატრიცულად გამრავლდება G დამშიფრავ მატრიცაზე. მიიღება დაშიფრული ორობითი ვექტორი. ხდება მათი, როგორც ორობითი რიცხვების, გადაყვანა ათობითში, ხოლო შემდეგ ათობითი რიცხვების, როგორც ASCII-კოდების, შესაბამისი სიმბოლოების მიღება. მიიღება დაშიფრული ტექსტი.

3. საწყისი ტექსტის აღდგენა.

მესამე ნაწილში ხდება დაშიფრული ტექსტიდან საწყისი ხექსტის აღდგენა. დაშიფრული ტექსტის სიმბოლოები გადაიყვანება ASCII-კოდებში, როგორც ვექტორები, მატრიცულად გამრავლდება H გამშიფრავ მატრიცაზე, მიღებული ორობითი რიცხვები გადაიყენება ათობითში და დაიწერება მათი შესაბამისი სიმბოლოები. აღდგება საწყისი ტექსტი.

მოცემული m რიცხვის საშუალებით 1,2, ... n რიცხვების გარკვეული გადანაცვლების იდენტიფიკაცია

m რიცხვების ფორმირება ხდება გადამცემისა და მიმღების მიერ გასაღებების ღია გაცვლის რომელიმე მეთოდით (მოდელში m რიცხვი კლავიატურიდან შეგვყავს).

1,2, ... n რიცხვების გადანაცვლებათა რაოდენობა არის n გადანაცვლებები შეიძლება დალაგდეს მიმდევრობით, ანუ გადაინომროს (დამყარდეს ურთიერთცალსახა შესაბამისობა 1,2, ... n! რისცვებსადა 1,2, ... n რიცხვების გადანაცვლებებს შორის).

მაგალითად: 1,2,3; 1,3,2; 2,1,3; 2,3,1; 3,1,2; 3,2,1 (თითოეული გადანაცვლება წარმოვადგინოთ ათობით რიცხვებად (123) და დავალაგოთ ზრდადობის მიხედვით).

ჩვენი მიზანია, მოცემული mრიცხვების სასუალებით მოვახდინოთ იმ გადანაცვლების ფორმირება, რომელიც მას შეესაბამება. ეს ხორციელდება შემდეგნაირად:

ჩავთვალოთ, რომ m მოთავსებულია 1-სა და n!-ს შორის (წინააღმდეგ შეთხვევაში m-ს ავიღებთ n!-ის მოდულით (0-ის ნაცვლად n!-ს ავიღებთ)).

[1,n!] შუალედი დავყოთ n ბლოკებად (მონაკვეთებად), რომელთაგან თითოეულის სიგრძეა (n-1)! (n(n-1)!=n!). ვნახოთ, რომელ ბლოკში ხვდება მოცემული m რიცხვი დაამ ბლოკის ნომერი დავწეროთ პირველ ადგილზე. შემდეგ დავამყაროთ შესაბამისობა 1,2, ... n-1 მიმდევროვისას და ზრდადობის მიხედვით დალაგებულ დარჩენილ (n-1) რიცხვს შორის

(დარჩენილი (n-1) რიცხვი გადავნომროთ). ვნახოთ mრიცხვის ნომერი თავის ბლოკში დაამ რიცხვს დავარქვათ m_1. ეს ბლოკი კიდევ დავყოთ (n-1) ბლოკებად, რომელთაგან თითოეულის სიგრძეა (n-2)! ((n-1)·(n-2)!=(n-1)!). ვნახოთ, რომელ ბლოკში ხვდება m_1 რიცხვი და ამ ბლოკის ნომრის შესაბამისი რიცხვი (გადანომრილი (n-1) რიცხვიდან) დავწეროთ მეორე ადგილზე. შემდეგ დარჩენილი (n-2) რიცხვი დავალაგოთ ზრდადობის მიხედვით და გადავნომროთ და ა.შ. სანამ არ დავწერთ ყველა რიცხვს. მიიღება m რიცხვის შედაბამისი გადანაცვლება.

ორობითი მიმდევრობის მიხედვით 1,2, ... n რიცხვების გარკვეული გადანაცვლების ფორმირება

თუ n რიცხვი დიდია, 1,2, ... n რიცხვების გადანაცვლების იდენტიფიკაცია ზემოთ მოყვანილი მეთოდით სირთულეებთანაა დაკავშირებული, რადგან მასში ოპერაციები ხორციელდება n! რიგის რიცხვებზე. ამის გამო, შესაძლებელია, უფრო მიზანშეწონილი იყოს ორობითი მიმდევრობის მიხედვით პირდაპირ 1,2, ... n რიცხვების გადანაცვლების ფორმირება. ეს შეიძლება შემდეგნაირად განხორციელდეს: გადამცემი და მიმღები ახდენენ ორობით მიმდევრობის ფორმირებას. ეს მიმდევრობა იყოფა თანაბარი სიგრძის ბლოკებად. ბლოკის სიგრძე ისე უნდა იყოს შერჩეული, რომ უდიდესი ორობითი რიცხვი 11...1 (რომლის თანრიგი ტოლი იქნება ბლოკის სიგრძის), გადაყვანილი ათობითში, მეტი ან ტოლი იყოს n-ის. გადანაცვლების ფორმირება ხდება შემდეგნაირად: ვიღებთ პირველ ბლოკს, გადაგვყავს ათობითში (თუ მეტია n-ზე, ვიღებთ n-ის მოდულით) და ვწერთ პირველ ადგილზე. ყოველი შედეგი რიცხვის აღებისას იმ შემთხვევაში, თუ ის უკვე იყო გამოყენებული, ვზრდით (ან ვამცირებთ) მას 1-ით და ისევ ვადარებთ გამოყენებულ რიცხვებს. თუ ისიც იყო გამოყენებული, ისევ 1-ით ვზრდით (ან ვამცირებთ) და ა.შ. სანამ

არ მივიღებთ ისეთ რიცხვს, რომელიც გამოყენებული არ ყოფილა. ამრიგად შესაძლებელია [1;n] შუალედის ყოველი რიცხვის მიღება, ანუ H მატრიცის სტრიქონების გადანომვრა და შესაბამისად მისი ფორმირება.